INTERNATIONAL ATOMIC ENERGY AGENCY
UNITED NATIONS EDUCATIONAL, SCIENTIFIC AND CULTURAL ORGANIZATION

INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS
34100 TRIESTE (ITALY) - P.O.B. 586 - MIRAMARE - STRADA COSTIERA 11 - TELEPHONES: 224281/2/3/4/5/6
CABLE: CENTRATOM - TELEX 460392 - I

SMR.201/3

SECOND WORKSHOP ON MATHEMATICS IN INDUSTRY

(2 - 27 February 1987)

"MODERN OPTIMIZATION SOFTWARE."

G.A. EVANS
Department of Mathematics
University of Technology
Loughborough, Leics. LE11 3TU
U.K.

CONTENTS

I. CLASSIFICATION OF THE PROBLEM

1. Introduction

Before attempting to solve a given optimization problem in practical terms, either by choosing a suitable library routine or by writing a routine, the choice of method needs to be made. The problem itself can be classified, to lead to a particular subset of methods from which the final choice is made. This choice requires balancing many criteria usually of a conflicting nature. For example, a very sophisticated routine may prove too large for a given system whereas a simple routine may prove too slow. A sophisticated routine may require considerable input by the user to control the many 'tuning variables' which require inside knowledge. Without these variables being well chosen the routine may loose some of its competitiveness.

Even when a routine is chosen there are many further problems which need consideration. Clearly any results which have been obtained need to be treated with caution. It is always advisable to treat critically any numbers thrown out by a computer. Traditional optimization routines deliver a local minimum which may not be the required minimum. In a many variable problem the hypersurface being minimized can have the most complex structure. Finding a global minimum is not an easy problem. Usually physical considerations can eliminate an unexpected extra local minimum, and some further computational experiments can resolve the shape of the function being treated, though this is not easy for a problem of high dimensionality.

2. Types of problem

The main split in classifying problems is to separate constrained problems from unconstrained. For unconstrained problems we have the formulation:

$$\text{minimize } F(\underline{x})$$

where F is a real valued function of the vector $\underline{x} \in E^n$ where

$$\underline{x} = (x_1, x_2, \ldots x_n)^T .$$

This is the general unconstrained problem in n variables.

Special methods are available for various sub-classes of this problem. If $n = 1$ a whole set of robust methods, line search methods, may be used. For n above say 10 special techniques are employed. Whether or not it is feasible to calculate analytically first derivatives or second derivatives of F gives a further breakdown in method choice. A further problem classification depends on any special nature which F may possess. For example, F could be linear, sums of squares of linear functions, quadratic, sums of squares of non-linear functions or just plain non-linear. Advantage can be taken of these classes of objective functions. It is generally true that the more information on F which can be utilized, the more effective the optimization becomes.

Constrained problems also have a classification due to the objective function, and in addition may be classified according to constraint. Constraints may be classified as equality constraints such as

$$c_i(\underline{x}) = 0 \qquad i = 1, \ldots, m_1$$

as inequality constraints such as

$$c_i(x) \geq 0 \qquad i = m_1+1, \ldots, m_2$$

and as range constaints such as

$$v_j \leq c_i(x) \leq w_j \qquad i = m_2+1_1, \ldots, m_3$$

$$j = 1, \ldots, m_3 - m_2 .$$

Within this classification the constraints may be linear or non-linear. The classical methods of linear programming would for example be used for a linearly constrained, linear function $F(\underline{x})$.

II CLASSIFICATION OF THE METHODS: UNCONSTRAINED

1. Univariate methods

The above analysis of the type of problem reflects the choice of method. For a one dimensional unconstrained problem the following algorithms may be considered:

(i) Fibonacci Search: This is a robust, simple interval reducing method based on being able to compute $F(\underline{x})$ at two interior points [1] p. 84.

(ii) Golden section search: This method has a constant reduction of the interval of uncertainty by the factor 0.6180. [1] p. 91.

(iii) Polynomial interpolation: Often used as a line search in multivariable algorithms such as Powell (quadratic), Davidon (cubic)[3]. [line searches of example 2 (v)].

(iv) Rational approximation: Using the ratio of two polynomials gives some advantages away from the minimum[16]. [line search of example 2 (vii)].

(v) Safeguarded methods These are based on (iii) and (iv) but check for locally invalid results, by returning to a more robust method which would not necessarily converge at the same rapid rate near the minimum.

2. Multivariate methods

For these methods there are many algorithms and each has many variations. For example many require internal one-dimensional line searches to be carried out, and the sort of choice above becomes available. Methods in this section include:

(i) Direct search methods: These are based on function comparisons at a sequence of points. The two most commonly used methods are those of Rosenbrock[5], [Example 2 (iii)], and the simplex or polytype algorithm[6], [Example 2 (iv)]. These methods are very crude, require the setting of many tuning parameters and should be used only where the objective function or its derivative exhibit discontinuities in the region of interest. For smooth functions, a knowledge of, or an ability to estimate numerically, derivatives of F will usually result in a far more satisfactory approach.

(ii) Newton's method: This is discussed elsewhere and is based on a local quadratic model of F obtained by the first three terms of a Taylor expansion and requires the Hessian matrix $\underline{\underline{G}}$ that is second derivatives of F. The search direction is defined by $\underline{p}$ and is given by

$$\bar{\underline{\underline{G}}}\,\underline{p} = -\underline{g}.$$

where $\underline{g}$ is the gradient vector and $\bar{\underline{\underline{G}}}$ is a positive definite matrix related to the true Hessian. This related matrix is found by decomposition of $\underline{\underline{G}}$ by either spectral or Cholesky [7] means and then by suitable reformulation to establish positive definitenss.

(iii) If estimates of the Hessian matrix are found by finite differences of $\underline{g}$ then an approximate Newton method arises. This is usually called a discrete Newton Method. These have many advantages over Newton being good at local convergence, but require many more gradient values to be computed.

(iv) Quasi-Newton methods or variable metric methods:[8] [Example 2 (v)]. These methods rely on building up information on $\underline{\underline{G}}$ from previous iterations, rather than computing $\underline{\underline{G}}$ (or an estimate) at a single point. The required matrices do not require the explicit formulation of the Hessian matrix. Some very general families of update formulae exist based on Broyden's family[10], [4]. They result in a sequence of Hessian estimates B_k with free parameters which can be chosen to enhance the efficiency of the methods in certain cases. It has been proved that if exact line searches are made many variations of the quasi-Newton methods yield identical points[9]. In practise the accuracy of line searches needs careful consideration and may introduce significant differences in the sequences of points associated by different algorithms.

(v) Non-derivative quasi-Newton methods: These rely on estimating first and second derivatives, but require more than first finite difference replacements of the exact derivatives. Such a procedure can destroy the logic of a quasi-Newton method and safeguards have to be added to the programme. Again estimating derivatives increases the number of function evaluations considerably.

(vi) Special methods: These include many recent attempts at modelling the problem with say multi-dimensional polynomials (Winfield)[15] and rational functions (Storey)[16]. These methods can be very effective for low dimensionality.

(vii) Conjugate gradient methods: The search directions are now not based on a stored matrix and would be used if a matrix became too large. Powell's[2] [Examples 1 (ii) and 2(vi)] method is based on a quadratic model which generates conjugate directions. A line search is then applied along each direction and an updated set of directions generated for the next step. These algorithms seem to excell when there is a large dimensionality in the problem.

III CLASSIFICATION OF METHODS - CONSTRAINED

The general constrained problem is considerably more involved than the linearly constrained problem as movement in any direction could alter the value of the constraint. In the linear case a search direction can be computed to satisfy the constraint automatically at all trial points of the current iteration. General methods include

(i) Penalty function methods[11]: These rely on constructing a new object function which has constructed large values in the region of non-feasibility. An unconstrained method can then be used directly on the new function.

(ii) Barrier function methods[11]: Penalty function methods do allow iterates to enter the non-feasible region. Barrier functions are so designed to introduce a positive singularity on the boundary of the feasible region. In this way non-feasible points are not permitted.

(iii) Lagrange multipliers: This is the classical approach to equality constraints and by using active sets. Inequality constraints may be also implemented. A suitably penalty term can be included using an augmentation to the normal Lagrangian[12].

(iv) Reduced gradient methods: Here the constraints are used to fix a search direction at each step which reflects the reduction in the number of degrees of freedom which arise from the constraints. These methods appear to function well for nearly all linear constraints. For non-linear constraints the steps may prove impracti-cally small. For inequality constraints the active set strategy is employed in which some of the inequalities are on their limits and can be treated as equalities. The search then moves along the boundary of the inequality constaint in question.

(v) Simple bounds: A bound constraint is said to be active if its variable is equal to one of the bounds. Hence an iteration is generated in which the working set is partitioned into free components and fixed components. Only the free components are then pursued in an essentially unconstrained problem.

IV METHODS FOR SUMS OF SQUARES

This is a commonly used special case and advantage can be made of the special property of the objective function. Problems of this type can arise in non-linear parameter estimation and in solving a set of non-linear equations by minimizing the sums of their squares. Hence if

$$F(x) = \frac{1}{2}\sum_{i=1}^{m} f_i(x)^2$$

then if the Jacobian matrix of $\underline{f}$ is $\underline{\underline{J}}$ and if $\underline{\underline{G}}_i(x)$ is the Hessian of $f_i(x)$ then

$$\underline{g}(x) = \underline{\underline{J}}(x)^T\underline{f}(x)$$

and

$$\underline{\underline{G}}(x) = \underline{\underline{J}}(x)^T\underline{\underline{J}}(x) + \sum_{i=1}^{m} f_i(x)G_i(x)$$

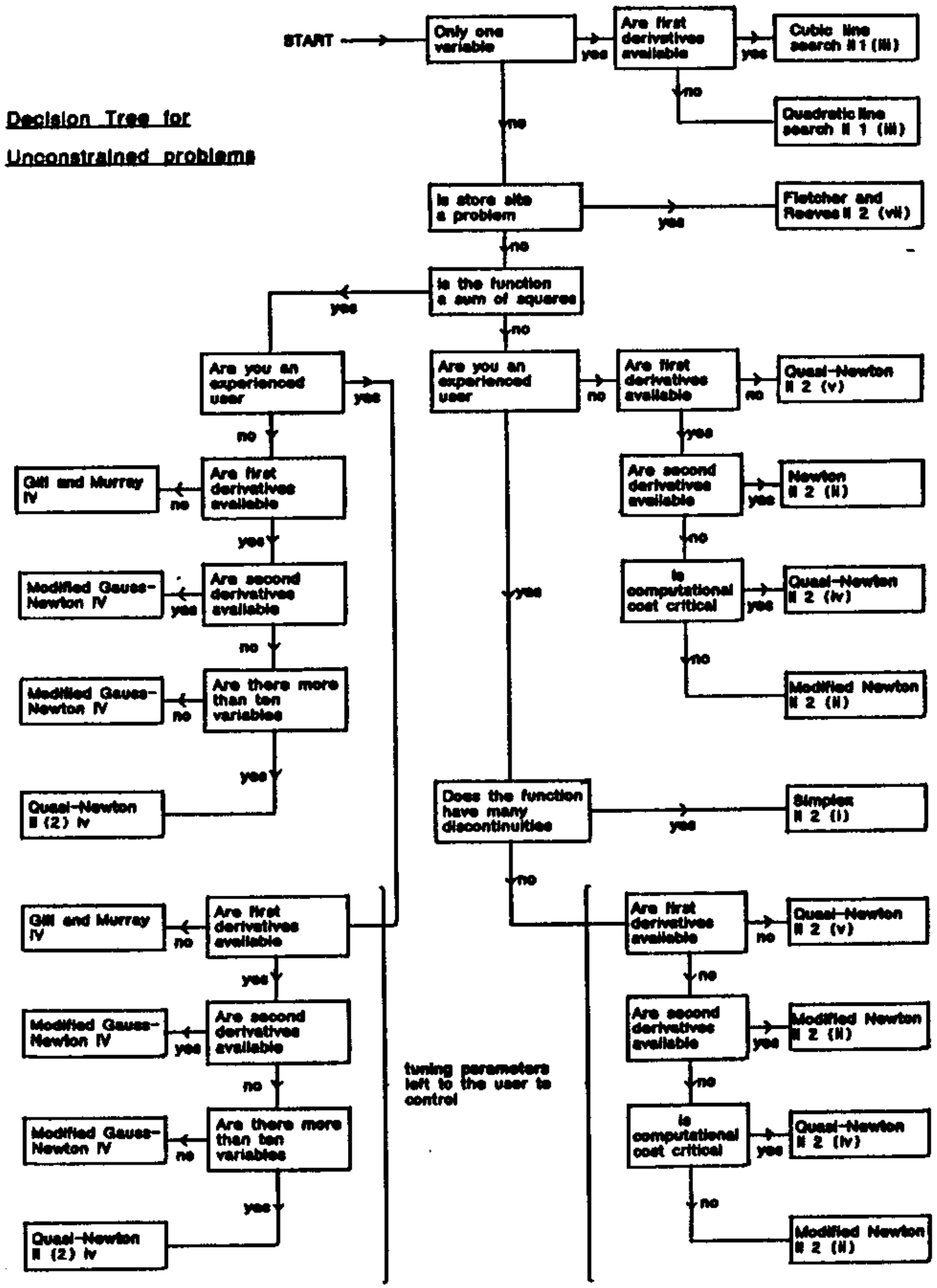
The eventual domination of $\underline{\underline{J}}(x)^T\underline{\underline{J}}(x)$ over the second order term is used to advantage. This yields the method called the Gauss-Newton method. Gill and Murray have a modification of this method if derivatives are not available.

V DECISION TREE

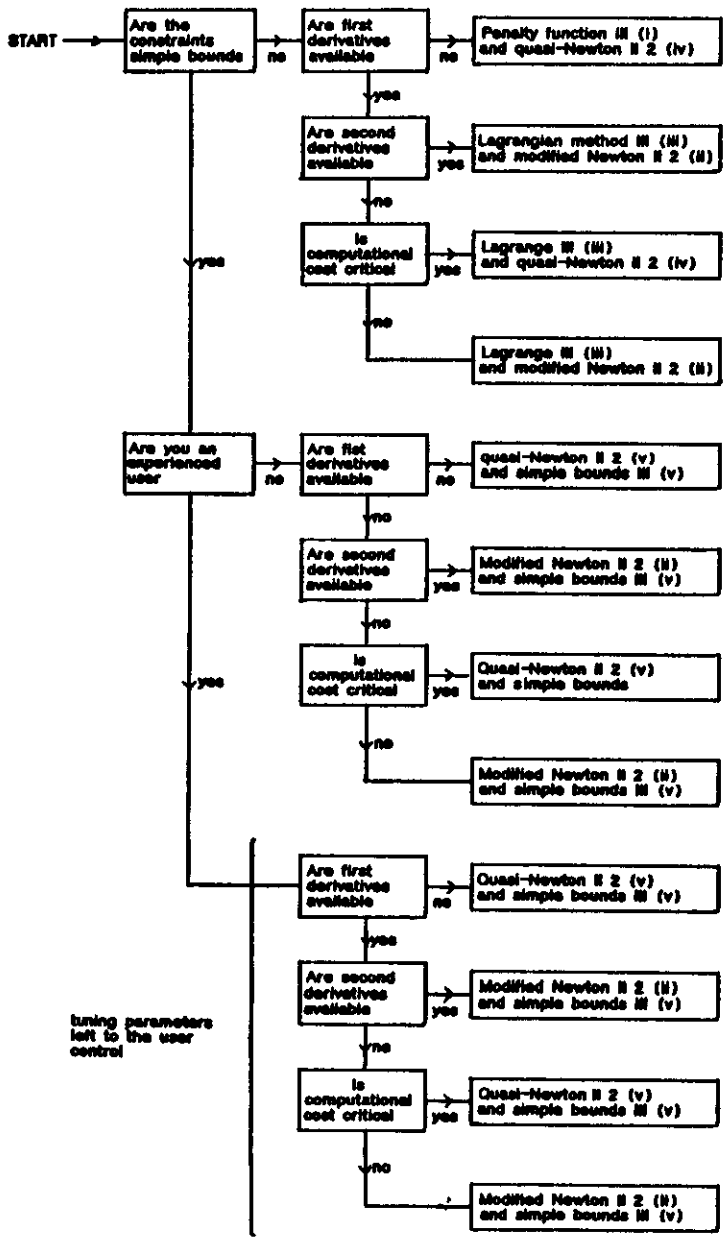
The result of sections II, III and IV is that a decision tree can be made up to guide prospective library users to a suitable routine. By having one version of a method in which average values of tuning parameters are used, a deep knowledge of the techniques by the user is balanced against the gain in efficiency which would follow by a good choice of these parameters.

Typical trees are shown below:

Decision Tree for Unconstrained problems



Decision Tree for constrained problems



VI CODING DETAILS

1. Pre-processors A number of generally applicable codes employ pre-processors to assess the problem in a crude fashion so as to fix 'tuning parameters' automatically, and often produce a good starting point for the main algorithm. This may involve little more than a grid of function evaluations to yield information on the rates of change of derivatives, the sizes of the objective function and the possibilities of close-spaced minima.

2. Line searches

These are carried out along a given direction in most multivariate methods and commonly a cubic fit using gradients is used if gradients are available. If not a quadratic fit is employed. More recently attempts at rational fitting have proved encouraging but harder to implement. The main coding problem is to determine the accuracy to which line searches should be made. A very stringent convergence criterion will increase the number of iterations for the line searches but decrease the number of steps in the main algorithm. Problems arise if the local object function is either very steep with respect to the distance along the optimizing direction or else very flat. From a Taylor expansion it is clear that in one dimension

$$f(x) = f(x_0) + (x - x_0)f'(x_0) + \frac{(x - x_0)^2}{2} f''(x_0) + \ldots$$

so that near the minimum

$$f(x) - f(x_0) = \frac{(x - x_0)^2}{2} f''(x_0) + \ldots$$

and hence the accuracy in f is equivalent to the square of the accuracy in x if $f''(x_0) \neq 0$. Hence only half of machine accuracy can be expected in x, or less if $f''(x_0)$ is zero[17]. The line search has to be terminated when any one of the following conditions has been triggered:

(1) The step size has passed some pre-determined limit. This guards against directions along which f is unbounded below.

(2) $|f_{k+1} - f_k| < \epsilon (|f_k| + \delta)$ where f_k, f_{k+1} are successive f values. ϵ is just greater than machine accuracy. δ is a variable to allow for f to be close to zero. Hence there is an automatic switch from relative to absolute errors in this case.

(3) $|x_{k+1} - x_k| < \sqrt{\epsilon} (|x_k| + \delta)$. This condition implies $f''(x_k) \neq 0$. An error message could be relayed if this fails to be satisfied after (2) has triggered. (This would probably imply a high order local supremum). Some algorithms are more tolerant of inaccurate line searches than others Dixon[13], but over stringency can cause the line search to get in a closed loop. An escape clause based on a maximum number of function evaluations is useful in these cases.

3. Stopping criteria

The above conditions for stopping a line search will also apply to the main algorithm. Now $\underline{x}$ is a vector and condition (3) must hold for each variable. Many pitfalls arise if a surface flattens in some region and then becomes more steep. It is possible to 'fool' the stopping criteria. Safeguards include restarting the algorithm at a sprinkling of points round the first minimum found, or even trying another method.

4. Scaling

Theoretically algorithms such as Newton's method are scale invariant with respect to linear transformations under exact arithmetic. However floating point arithmetic is not scale-invariant and so variables of widely differing magnitudes can cause considerable problems. Estimating derivatives using finite differences can also cause numerical instability when close-valued, large numbers are subtracted with the inevitable loss of precision.

VII EXAMPLE CODES

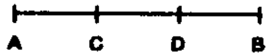
In this section some of the methods mentioned above are implemented. A range of implementations is covered from programes for a large main frame machine down to code suitable for the cheapest micro. It is an interesting exercise to consider just what can be implemented on a cheap small micro in terms of numerical algorithms. For example a quasi- Newton algorithm which requires substantial matrix work may not be feasible as the programme itself will absorb most of the storage. What is needed is a low storage requirement, possibly at the expense of run-time. With a micro an overnight run is not out of the question. Language is also not a problem as most of these programmes are straight forward 'number crunching' exercises, though modern methods based on artificial intelligence (to say automatically choose a method from a library, or to dynamically vary the tuning parameters) are being written. With this in mind Algol 68 has been used for main frame examples (and is easily converted to Pascal or PLI) and basic for micro work

1. Micro codes

(i) Golden section

One robust method for a single variable function depends on the geometric Golden section. The idea is to evaluate f(x) at four points. The outer pair at x = a and x = b bracket the minimum. The inner pair are chosen so that the ratio below holds.

$$\frac{CB}{AB} = \frac{CD}{DB}$$



Hence if the minimum lies between C and B, A can be rejected leaving an exactly similar (though scaled down) set of points CDB as the original set ACB. In this way only one new function evaluation is needed at each step and the minimum is robustly approached.

```
10 INPUT "Input bracketing bounds a and b", Chr$13; a,b
20 INPUT "Input accuracy", ep
30 LET k=(SQR(5)-1)/2
40 LET c=b-k*(b-a)
50 LET x1=a
60 GOSUB 1000
70 LET fa=f
80 LET x1=b
90 GOSUB 1000
100 LET fb=f
110 LET x1=c
120 GOSUB 1000
130 LET fc=f
140 LET d=a+b-c
145 PRINT a,b
```

```
146 IF ABS(a-b)<ep THEN STOP
150 LET x1=d
160 GOSUB 1000
170 LET fd=f
180 IF fc<fa AND fc<fd THEN GOTO 240
190 LET a=c: LET fa=fc: LET c=d: LET fc=fd: GOTO 140
240 LET b=d: LET fb=fd: LET d=c: LET fd=fc
280 LET c=a+b-d
290 LET x1=c: GOSUB 1000: LET fc=f
320 PRINT a,b
325 IF ABS(a-b)<ep THEN STOP
330 GOTO 180
1000 LET f=x1*x1+2*EXP(-x1)
1010 RETURN
```

For the example given inputting a bracketing interval 0 to 1 and accuracy 1.0e-4 gives the sequences for a and b of

0	1
0.38196601	1
0.38196601	0.76393202
0.38196601	0.61803399
⋮	⋮
0.567116	0.56718132

to give the required minimum at x = 0.5671 to four figures after 22 steps.

(ii) <u>Powell's method</u>. For functions of many variables we get into the realms of more serious numerical computing. Some very refined packages are available on main frame machines. One of these is due to Powell and involves a sequence of one dimensional optimizations along a set of orthogonal direction vectors (d) from the current starting point. After each one dimensional search is complete then direction vectors are adjusted so that the first of these vectors becomes a preferred direction for subsequent steps. The one dimensional part is written as subroutine 2000 and is based on fitting a quadratic to three points (a(1), a(2), a(3)) and locating the minimum of this quadratic. One of the original three points is replaced by the new estimate and the process repeats to give rapid convergence

One of the classic test functions for optimization routines is the two dimensional banana valley of Rosenbrock. The example causes considerable problems to many routines as the valley is steep sided and slopes gently from (-1,1) to a minimum at (1,1). This is coded at 4000.

```
10 DIM x(5):DIM y(6,5):DIM d(5,5):DIM f(6)
20 DIM z(5):DIM w(5):DIM u(5):DIM c(5)
30 DIM a(4):DIM b(4)
100 INPUT "Input number of variables":CHR$ 13:n
110 PRINT "Input the start vector for the search"
120 FOR I=1 TO n:INPUT x(I):NEXT I
140 FOR I=1 TO n:FOR J=1 TO n:LET d(I,J)=0
150 IF I=J THEN LET d(I,J)=1:NEXT J: NEXT I
200 INPUT "Input accuracy": CHR$ 13:ep
210 INPUT "Maximum allowed steps":CHR$ 13:h,hb
```

```
300 FOR i=1 TO n:LET y(1,i)=x(i):LET z(i)=x(i):NEXT i
320 GOSUB 4000
325 LET f(1)=ff
330 FOR r=1 TO n
340 FOR i=1 TO n:LET w(i)=y(r,i):NEXT i
370 FOR i=1 TO n:LET c(i)=d(r,i):NEXT i
400 GOSUB 2000
405 LET f(r+1)=ef:PRINT"y-values"
410 FOR i=1 TO n
420 LET y(r+1,i)=w(i)+lam*c(i)
425 PRINT y(r+1,i)
430 NEXT i
440 NEXT r
500 LET fm=f(1)-f(2)
510 LET im=1
520 FOR i=2 TO n
530 IF fm>f(i)-f(i+1) THEN GOTO 560
540 LET fm=f(i)-f(i+1)
550 LET im=i
560 NEXT i
570 LET qu=im:LET del=fm:LET f1=f(1):LET f2=f(n+1)
620 FOR i=1 TO n:LET z(i)=2*y(n+1,i)-y(1,i):NEXT i
650 GOSUB 4000
660 LET f3=ff:LET u1=f1-2*f2+f3
710 LET u2=f1-f2-del:LET u3=f1-f3
730 IF f3>=f1 OR u1*u2*u2>=del*u3*u3/2 THEN GOTO 1000
740 FOR i=1 TO n:LET w(i)=y(n+1,i):NEXT i
770 FOR i=1 TO n:LET c(i)=y(n+1,i)-x(i):NEXT i
800 GOSUB 2000
```



```
820 FOR i=qu TO n-1
830 FOR j=1 TO n
840 LET d(i,j)=d(i+1,j)
850 NEXT j:NEXT i
860 LET sm=0
870 FOR i=1 TO n
880 LET d(n,i)=y(n+1,i)-x(i)
885 IF ABS d(n,i)>sm THEN LET sm=ABS d(n,i)
890 NEXT i
895 FOR i=1 TO n:LET d(n,i)=d(n,i)/sm: NEXT i
900 FOR i=1 TO n:LET u(i)=w(i)+lam*c(i):NEXT i
930 GOTO 1025
1000 FOR i=1 TO n:LET u(i)=y(n+1,i):NEXT i
1025 PRINT "u-variables"
1030 FOR i=1 TO n: PRINT u(i): NEXT i
1060 FOR i=1 TO n
1070 LET x(i)=u(i):LET y(1,i)=u(i):LET z(i)=u(i)
1080 NEXT i
1120 GOSUB 4000
1140 IF ABS (ff-f(1))<ep+ep*10*(ABS(f(1))+.5) THEN STOP
1150 LET f(1)=ff
1160 GOTO 330
2000 FOR i=1 TO n
2010 LET z(i)=w(i)
2020 NEXT i
2030 GOSUB 4000
2040 LET a(1)=0:LET b(1)=ff
2045 LET cm=ABS c(1)
2050 FOR i=2 TO n:IF cm <ABS c(i) THEN LET cm=ABS c(i)
```

```
2052 NEXT I
2055 FOR I=1 TO n:LET c(I)=c(I)/cm:NEXT I
2060 FOR I=1 TO n:LET z(I)=w(I)+h*c(I):NEXT I
2090 GOSUB 4000
2100 LET a(2)=h:LET b(2)=ff
2120 IF b(1) <b(2) THEN GOTO 2180
2130 FOR I=1 TO n:LET z(I)=w(I)+2*h*c(I):NEXT I
2160 LET a(3)=2*h
2170 GOTO 2220
2180 FOR I=1 TO n:LET z(I)=w(I)-h*c(I):NEXT I
2210 LET a(3)=-h
2220 GOSUB 4000
2230 LET b(3)=ff
2240 LET u2=0
2245 IF a(1) <a(2) THEN GOTO 2305
2250 LET u1=a(1):LET a(1)=a(2):LET a(2)=u1
2280 LET u1=b(1):LET b(1)=b(2):LET b(2)=u1
2305 IF u2=1 THEN GOTO 2400
2310 IF a(2) <a(3) THEN GOTO 2380
2320 LET u1=a(2):LET a(2)=a(3):LET a(3)=u1
2350 LET u1=b(2):LET b(2)=b(3):LET b(3)=u1
2380 LET u2=1
2390 GOTO 2245
2400 LET u1=a(2)-a(3)
2402 PRINT "A","F"
2403 FOR I=1 TO 3: PRINT a(I),b(I):NEXT I
2407 CLS
2410 LET u2=a(3)-a(1)
2420 LET u3=a(1)-a(2)
```



```
2430 LET u4=u1*b(1)+u2*b(2)+u3*b(3)
2440 LET lm=(u1*(a(2)+a(3))*b(1)+u2*(a(3)+a(1))*b(2)
     +u3*(a(1)+a(2))*b(3))/u4/2
2445 PRINT "lm=",lm
2450 LET min=u4<u3*u1*u2
2455 PRINT "MIN=",min
2460 LET u1=ABS (lm-a(3))
2470 LET md=u1
2480 IF ABS(lm-a(1)) <u1 THEN LET md=ABS(lm-a(1))
2490 IF min=0 OR md>hb THEN GOTO 2850
2500 FOR I=1 TO n:LET z(I)=w(I)+lm*c(I):NEXT I
2530 GOSUB 4000
2531 PRINT lm,ff
2540 LET md=ABS(lm-a(1))
2550 LET ef=b(1)
2560 LET lam=a(1)
2570 FOR I=2 TO 3
2580 IF md < ABS(lm-a(I)) THEN GOTO 2620
2590 LET md=ABS(lm-a(I))
2600 LET ef=b(I)
2610 let lam=a(I)
2620 NEXT I
2630 IF md>ep THEN GOTO 2700
2640 IF ff>ef THEN RETURN
2650 LET lam=lm:LET ef=ff:RETURN
2700 IF lm>a(1) AND lm <a(2) AND ff <b(1) AND ff <b(2)
     THEN GOTO 2770
2710 IF lm>a(2) AND lm <a(3) AND ff <b(2) AND ff <b(3)
     THEN GOTO 2790
```

```
2730 LET ir=1
2740 IF b(2)>b(1) AND b(2)>b(3) THEN LET ir=2
2750 IF b(3)>b(1) AND b(3)>b(2) THEN LET ir=3
2760 GOTO 2800
2770 LET ir=3:GOTO 2800
2790 LET ir=1
2800 LET a(ir)=lm:LET b(ir)=ff:GOTO 2240
2830 PRINT "ff is largest f value":RETURN
2850 IF b(l) <b(3) THEN GOTO 2930
2860 LET a(1)=a(3)+hb
2870 FOR i=1 TO n:LET z(i)=w(i)+a(1)*c(i):NEXT i
2910 GOSUB 4000
2920 LET b(1)=ff
2925 GOTO 2240
2930 LET a(3)=a(1)-hb
2940 FOR i=1 TO n:LET z(i)=w(i)+a(3)*c(i):NEXT i
2970 GOSUB 4000
2980 LET b(3)=ff
2990 GOTO 2240
4000 LET fu=z(2)-z(1)*z(1)
4010 LET fv=1-z(1)
4020 LET ff=100*fu*fu*fv*fv
4030 RETURN
```

The full output includes intermediate points for each line search. On the famous 'banana' function shown above (4000) the points reached in the major steps are:

z(1)	z(2)	f
-1.2	1.0	24.2
-0.9896	0.9896	3.969
-0.7810	0.5397	3.256
-0.5055	0.2062	2.5097
-0.2249	0.0046	1.7095
0.0756	-0.02919	0.9763
0.3292	0.07928	0.53456
0.5645	0.2972	0.2357
0.7656	0.5719	0.07507
0.9256	0.8502	0.0098821
0.9951	0.9914	0.0001567
1.00000	1.00000	2.2E-10

An accuracy of 1.0E-4 and allowed steps of 0.5 and 2.5 were used.

2. <u>Main frame codes</u>

(i) An Algol 68 version of the golden section algorithm appears below and is tested on the range of functions of MODE AP. Note the delicate stopping criteria.

```
PROGRAM exgoldsec
 BEGIN
    MODE  AP =[1:5] PROC ( REAL ) REAL ;
     AP ff;
    [1:5] REAL xl,xu;

    PROC cosh=(REAL x)REAL:
    BEGIN
       REAL w:=exp(x);
       (w+1.0/w)/2.0
    END;

     PROC golsec=( PROC ( REAL ) REAL f, REAL x0,x1,er) VOID :
     BEGIN
        REAL c=(sqrt(5.0)-1.0)/2.0;
        REAL xl:=x0,xu:=x1,xn,xm;
        REAL fl:=f(x0),fu:=f(x1),fn,fm;
        FORMAT fo1=$l"x"2x4(+d.8de+zd3x),l"f"2x4(+d.8de+zd3x)l$;
        INT ul;
       xn:=xl+c*(xu-xl);
       xm:=xu+xl-xn;
       fn:=f(xn);
       fm:=f(xm);
       ul:=50;
        TO ul WHILE ( ABS (fl-fn)>er*( ABS fl+1.0e-20))
                     OR ( ABS (fn-fm)>er*( ABS fn+1.0e-20))
                     OR ( ABS (fm-fu)>er*( ABS fu+1.0e-20))
       DO
        IF fm<fn THEN xu:=xn;fu:=fn;xn:=xm;fn:=fm;
                        xm:=xu+xl-xn;fm:=f(xm)
                   ELSE xl:=xm;fl:=fm;xm:=xn;fm:=fn;
                        xn:=xu+xl-xm;fn:=f(xn)
         FI ;
        printf((fo1,xl,xm,xn,xu,fl,fm,fn,fu))
       OD
     END ;

    ff[1]:=( REAL x) REAL :((1.0-x)*(1.0-x)-1.0);
    ff[2]:=( REAL x) REAL :( REAL w:=(1.0-x);
                             w:=w*w*w;w:=w*w;
                             w-1.0);
    ff[3]:=( REAL x) REAL :( REAL w:=(1.0-x);
                             w:=w*w*w*w*w;w:=w*w;
                             w-1.0);
    ff[4]:=( REAL x) REAL :(exp(-2.0*x)-exp(-x));
    ff[5]:=( REAL x) REAL :(cos(x)+cosh(x)-1.0);
     FOR i1 TO 5 DO xl[i1]:=(i1|0,0,0,0,-1);
                    xu[i1]:=(i1|2,2,2,2,1) OD ;
     FOR i1 TO 5 DO
       golsec(ff[i1],xl[i1],xu[i1],1.0e-7);
      TO 5 DO newline(stand out) OD
     OD
 END
  FINISH
```


ff[1]

x	+7.63932019e- 1	+1.23606798e+ 0	+1.52786404e+ 0	+2.00000000e+ 0
f	-9.44271907e- 1	-9.44271907e- 1	-7.21359558e- 1	+0.00000000e+ 0
x	+7.63932019e- 1	+1.05572808e+ 0	+1.23606798e+ 0	+1.52786404e+ 0
f	-9.44271907e- 1	-9.96894382e- 1	-9.44271907e- 1	-7.21359558e- 1
x	+7.63932019e- 1	+9.44271922e- 1	+1.05572808e+ 0	+1.23606798e+ 0
f	-9.44271907e- 1	-9.96894382e- 1	-9.96894382e- 1	-9.44271907e- 1
x	+9.44271922e- 1	+1.05572808e+ 0	+1.12461182e+ 0	+1.23606798e+ 0
f	-9.96894382e- 1	-9.96894382e- 1	-9.84471895e- 1	-9.44271907e- 1
x	+9.44271922e- 1	+1.01315567e+ 0	+1.05572808e+ 0	+1.12461182e+ 0
f	-9.96894382e- 1	-9.99826930e- 1	-9.96894382e- 1	-9.84471895e- 1
x	+9.44271922e- 1	+9.86844331e- 1	+1.01315567e+ 0	+1.05572808e+ 0
f	-9.96894382e- 1	-9.99826930e- 1	-9.99826930e- 1	-9.96894382e- 1
x	+9.86844331e- 1	+1.01315567e+ 0	+1.02941674e+ 0	+1.05572808e+ 0
f	-9.99826930e- 1	-9.99826930e- 1	-9.99134652e- 1	-9.96894382e- 1
x	+9.86844331e- 1	+1.00310540e+ 0	+1.01315567e+ 0	+1.02941674e+ 0
f	-9.99826930e- 1	-9.99990359e- 1	-9.99826930e- 1	-9.99134652e- 1
x	+9.86844331e- 1	+9.96894598e- 1	+1.00310540e+ 0	+1.01315567e+ 0
f	-9.99826930e- 1	-9.99990359e- 1	-9.99990359e- 1	-9.99826930e- 1
x	+9.96894598e- 1	+1.00310540e+ 0	+1.00694486e+ 0	+1.01315567e+ 0
f	-9.99990359e- 1	-9.99990359e- 1	-9.99951772e- 1	-9.99826930e- 1
x	+9.96894598e- 1	+1.00073406e+ 0	+1.00310540e+ 0	+1.00694486e+ 0
f	-9.99990359e- 1	-9.99999464e- 1	-9.99990359e- 1	-9.99951772e- 1
x	+9.96894598e- 1	+9.99265937e- 1	+1.00073406e+ 0	+1.00310540e+ 0
f	-9.99990359e- 1	-9.99999464e- 1	-9.99999464e- 1	-9.99990359e- 1
x	+9.99265939e- 1	+1.00073406e+ 0	+1.00163728e+ 0	+1.00310540e+ 0
f	-9.99999464e- 1	-9.99999464e- 1	-9.99997318e- 1	-9.99990359e- 1
x	+9.99265939e- 1	+1.00016916e+ 0	+1.00073406e+ 0	+1.00163728e+ 0
f	-9.99999464e- 1	-9.99999970e- 1	-9.99999464e- 1	-9.99997318e- 1

ff[2]

x	+9.99265939e- 1	+9.99830842e- 1	+1.00016916e+ 0	+1.00073406e- 0
f	-9.99999464e- 1	-9.99999970e- 1	-9.99999970e- 1	-9.99999464e- 1
x	+9.99830842e- 1	+1.00016916e+ 0	+1.00039575e+ 0	+1.00073406e- 0
f	-9.99999970e- 1	-9.99999970e- 1	-9.99999844e- 1	-9.99999464e- 1
x	+9.99830842e- 1	+1.00005743e+ 0	+1.00016916e+ 0	+1.00039575e+ 0
f	-9.99999970e- 1	-1.00000000e+ 0	-9.99999970e- 1	-9.99999844e- 1
x	+9.99830842e- 1	+9.99942571e- 1	+1.00005743e+ 0	+1.00016916e+ 0
f	-9.99999970e- 1	-1.00000000e+ 0	-1.00000000e+ 0	-9.99999970e- 1
x	+7.63932019e- 1	+1.23606798e+ 0	+1.52786404e+ 0	+2.00000000e+ 0
f	-9.99826930e- 1	-9.99826930e- 1	-9.78366219e- 1	+0.00000000e+ 0
x	+7.63932019e- 1	+1.05572808e+ 0	+1.23606798e+ 0	+1.52786404e+ 0
f	-9.99826930e- 1	-9.99999970e- 1	-9.99826930e- 1	-9.78366219e- 1
x	+7.63932019e- 1	+9.44271922e- 1	+1.05572808e+ 0	+1.23606798e+ 0
f	-9.99826930e- 1	-9.99999970e- 1	-9.99999970e- 1	-9.99826930e- 1
x	+9.44271922e- 1	+1.05572808e+ 0	+1.12461182e+ 0	+1.23606798e+ 0
f	-9.99999970e- 1	-9.99999970e- 1	-9.99996252e- 1	-9.99826930e- 1
x	+9.44271922e- 1	+1.01315567e+ 0	+1.05572808e+ 0	+1.12461182e+ 0
f	-9.99999970e- 1	-1.00000000e+ 0	-9.99999970e- 1	-9.99996252e- 1
x	+9.44271922e- 1	+9.86844331e- 1	+1.01315567e+ 0	+1.05572808e+ 0
f	-9.99999970e- 1	-1.00000000e+ 0	-1.00000000e+ 0	-9.99999970e- 1

ff[3]

x	+7.63932019e- 1	+1.23606798e+ 0	+1.52786404e+ 0	+2.00000000e+ 0
f	-9.99999464e- 1	-9.99999464e- 1	-9.98320341e- 1	+0.00000000e+ 0
x	+7.63932019e- 1	+1.05572808e+ 0	+1.23606798e+ 0	+1.52786404e+ 0
f	-9.99999464e- 1	-1.00000000e+ 0	-9.99999464e- 1	-9.98320341e- 1
x	+7.63932019e- 1	+9.44271922e- 1	+1.05572808e+ 0	+1.23606798e+ 0
f	-9.99999464e- 1	-1.00000000e+ 0	-1.00000000e+ 0	-9.99999464e- 1
x	+9.44271922e- 1	+1.05572808e+ 0	+1.12461182e+ 0	+1.23606798e+ 0
f	-1.00000000e+ 0	-1.00000000e+ 0	-1.00000000e+ 0	-9.99999464e- 1
x	+1.05572808e+ 0	+1.12461182e+ 0	+1.16718423e+ 0	+1.23606798e+ 0
f	-1.00000000e+ 0	-1.00000000e+ 0	-9.99999985e- 1	-9.99999464e- 1
x	+1.05572808e+ 0	+1.09830049e+ 0	+1.12461182e+ 0	+1.16718423e+ 0
f	-1.00000000e+ 0	-1.00000000e+ 0	-1.00000000e+ 0	-9.99999985e- 1

ff[4]

x	+0.00000000e+ 0	+4.72135961e- 1	+7.63932019e- 1	+1.23606798e+ 0
f	+0.00000000e+ 0	-2.34706052e- 1	-2.48832490e- 1	-2.06119942e- 1
x	+4.72135961e- 1	+7.63932019e- 1	+9.44271922e- 1	+1.23606798e+ 0
f	-2.34706052e- 1	-2.48832490e- 1	-2.37670710e- 1	-2.06119942e- 1
x	+4.72135961e- 1	+6.52475864e- 1	+7.63932019e- 1	+9.44271922e- 1
f	-2.34706052e- 1	-2.49569234e- 1	-2.48832490e- 1	-2.37670710e- 1
x	+4.72135961e- 1	+5.83592117e- 1	+6.52475864e- 1	+7.63932019e- 1
f	-2.34706052e- 1	-2.45648659e- 1	-2.49569234e- 1	-2.48832490e- 1
x	+5.83592117e- 1	+6.52475864e- 1	+6.95048273e- 1	+7.63932019e- 1
f	-2.45648658e- 1	-2.49569234e- 1	-2.49999098e- 1	-2.48832490e- 1
x	+6.52475864e- 1	+6.95048273e- 1	+7.21359611e- 1	+7.63932019e- 1
f	-2.49569234e- 1	-2.49999098e- 1	-2.49806538e- 1	-2.48832490e- 1
x	+6.52475864e- 1	+6.78787202e- 1	+6.95048273e- 1	+7.21359611e- 1
f	-2.49569234e- 1	-2.49947701e- 1	-2.49999098e- 1	-2.49806538e- 1
x	+6.78787202e- 1	+6.95048273e- 1	+7.05098540e- 1	+7.21359611e- 1
f	-2.49947701e- 1	-2.49999098e- 1	-2.49964714e- 1	-2.49806538e- 1
x	+6.78787202e- 1	+6.88837469e- 1	+6.95048273e- 1	+7.05098540e- 1
f	-2.49947701e- 1	-2.49995336e- 1	-2.49999098e- 1	-2.49964714e- 1
x	+6.88837469e- 1	+6.95048273e- 1	+6.98887736e- 1	+7.05098540e- 1
f	-2.49995336e- 1	-2.49999098e- 1	-2.49991808e- 1	-2.49964714e- 1
x	+6.88837469e- 1	+6.92676932e- 1	+6.95048273e- 1	+6.98887736e- 1
f	-2.49995336e- 1	-2.49999940e- 1	-2.49999098e- 1	-2.49991808e- 1
x	+6.88837469e- 1	+6.91208810e- 1	+6.92676932e- 1	+6.95048273e- 1
f	-2.49995336e- 1	-2.49999058e- 1	-2.49999940e- 1	-2.49999098e- 1
x	+6.91208810e- 1	+6.92676932e- 1	+6.93580151e- 1	+6.95048273e- 1
f	-2.49999058e- 1	-2.49999940e- 1	-2.49999952e- 1	-2.49999098e- 1
x	+6.92676932e- 1	+6.93580151e- 1	+6.94145054e- 1	+6.95048273e- 1
f	-2.49999940e- 1	-2.49999952e- 1	-2.49999752e- 1	-2.49999098e- 1
x	+6.92676932e- 1	+6.93241835e- 1	+6.93580151e- 1	+6.94145054e- 1
f	-2.49999940e- 1	-2.49999996e- 1	-2.49999952e- 1	-2.49999752e- 1
x	+6.92676932e- 1	+6.93015248e- 1	+6.93241835e- 1	+6.93580151e- 1
f	-2.49999940e- 1	-2.49999993e- 1	-2.49999996e- 1	-2.49999952e- 1
x	+6.93015248e- 1	+6.93241835e- 1	+6.93353564e- 1	+6.93580151e- 1
f	-2.49999993e- 1	-2.49999996e- 1	-2.49999991e- 1	-2.49999952e- 1
x	+6.93015248e- 1	+6.93126976e- 1	+6.93241835e- 1	+6.93353564e- 1
f	-2.49999993e- 1	-2.50000000e- 1	-2.49999996e- 1	-2.49999991e- 1

x	-2.36067981e- 1	+2.36067981e- 1	+5.27864039e- 1	+1.00000000e+ 0
f	+1.00025882e+ 0	+1.00025882e+ 0	+1.00647035e+ 0	+1.08338295e+ 0
x	-2.36067981e- 1	+5.57280779e- 2	+2.36067981e- 1	+5.27864039e- 1
f	+1.00025882e+ 0	+1.00000080e+ 0	+1.00025882e+ 0	+1.00647035e+ 0
x	-2.36067981e- 1	-5.57280779e- 2	+5.57280779e- 2	+2.36067981e- 1
f	+1.00025882e+ 0	+1.00000080e+ 0	+1.00000080e+ 0	+1.00025882e+ 0
x	-5.57280779e- 2	+5.57280779e- 2	+1.24611925e- 1	+2.36067981e- 1
f	+1.00000080e+ 0	+1.00000080e+ 0	+1.00002009e+ 0	+1.00025882e+ 0
x	-5.57280779e- 2	+1.31556690e- 2	+5.57280779e- 2	+1.24611825e- 1
f	+1.00000080e+ 0	+9.99999993e- 1	+1.00000080e+ 0	+1.00002009e+ 0
x	-5.57280779e- 2	-1.31556690e- 2	+1.31556690e- 2	+5.57280779e- 2
f	+1.00000080e+ 0	+9.99999993e- 1	+9.99999993e- 1	+1.00000080e+ 0
x	-1.31556690e- 2	+1.31556690e- 2	+2.94167399e- 2	+5.57280779e- 2
f	+9.99999993e- 1	+9.99999993e- 1	+1.00000006e+ 0	+1.00000080e+ 0
x	-1.31556690e- 2	+3.10540199e- 3	+1.31556690e- 2	+2.94167399e- 2
f	+9.99999993e- 1	+1.00000001e+ 0	+9.99999993e- 1	+1.00000006e+ 0

(ii) <u>Rosenbrock 1-d search</u>: This is a very easily coded line search method and is again illustrated in five examples.

```
PROGRAM exrosen
 BEGIN
    MODE  AP =[1:5] PROC ( REAL ) REAL ;
    AP ff;
   [1:5] REAL xar;

   PROC cosh=(REAL x)REAL:
   BEGIN
      REAL w:=exp(x);
      (w+1.0/w)/2.0
   END;

    PROC rosen=( PROC ( REAL ) REAL f, REAL e,er,x0) VOID :
    BEGIN
       INT ul;
       REAL xa:=x0,es:=e,fa,fb;
       BOOL b:= FALSE ;
       FORMAT fo1=$l"x="2x+d.8de+zd,10x"f="2x+d.8de+zd$;
      fa:=f(xa);
      ul:=50;
      fb:=fa+1.0;
       TO ul WHILE  ABS (fa-fb)>er*( ABS fa+1.0e-20) DO
         (b!fa:=fb;b:= FALSE );
         xa PLUSAB es;
         fb:=f(xa);
         printf((fo1,xa,fb));
          IF fb<fa THEN
            es TIMESAB 3;b:= TRUE
          ELSE xa MINUSAB es;es DIVAB -2
          FI
       OD
    END ;

   ff[1]:=( REAL x) REAL :((1.0-x)*(1.0-x)-1.0);
   ff[2]:=( REAL x) REAL :( REAL w:=(1.0-x);
                            w:=w*w*w;w:=w*w;
                            w-1.0);
   ff[3]:=( REAL x) REAL :( REAL w:=(1.0-x);
                            w:=w*w*w*w*w;w:=w*w;
                            w-1.0);
   ff[4]:=( REAL x) REAL :(exp(-2.0*x)-exp(-x));
   ff[5]:=( REAL x) REAL :(cos(x)+cosh(x)-1.0);
    FOR i1 TO 5 DO xar[i1]:=(i1!0,0,0,0,1) OD ;
    FOR i1 TO 5 DO
      (rosen(ff[i1],0.1,1.0e-7,xar[i1]);
        TO 5 DO newline(stand out) OD )
OD  END
FINISH
```

ff[1]


```
x=  +9.99999996e- 2          f=  -1.90000001e- 1
x=  +4.00000002e- 1          f=  -6.40000001e- 1
x=  +1.30000001e+ 0          f=  -9.09999996e- 1
x=  +4.00000006e+ 0          f=  +8.00000035e+ 0
x=  -4.99999672e- 2          f=  +1.02499931e- 1
x=  +1.97500005e+ 0          f=  -4.93746952e- 2
x=  +9.42500058e- 1          f=  -9.98593755e- 1
x=  -4.99990449e- 2          f=  +1.02499892e- 1
x=  +1.46875006e+ 0          f=  -7.80273379e- 1
x=  +7.09375061e- 1          f=  -9.15537141e- 1
x=  +1.08906257e+ 0          f=  -9.92067859e- 1
x=  +8.99218820e- 1          f=  -9.89843152e- 1
x=  +9.94140700e- 1          f=  -9.99965468e- 1
x=  +1.08906257e+ 0          f=  -9.92067859e- 1
x=  +9.46579763e- 1          f=  -9.97156955e- 1
x=  +1.01787117e+ 0          f=  -9.99680623e- 1
x=  +9.82275464e- 1          f=  -9.99685839e- 1
x=  +1.00007331e+ 0          f=  -9.99999993e- 1
x=  +1.01787117e+ 0          f=  -9.99680623e- 1
x=  +9.91174385e- 1          f=  -9.99922112e- 1
x=  +1.00452277e+ 0          f=  -9.99979548e- 1
x=  +9.97848585e- 1          f=  -9.99995373e- 1
x=  +1.00118569e+ 0          f=  -9.99998592e- 1
x=  +9.99517124e- 1          f=  -9.99999760e- 1
x=  +1.00035140e+ 0          f=  -9.99999873e- 1
x=  +9.99934271e- 1          f=  -9.99999993e- 1
```

ff[2]

```
x=  +9.99999996e- 2          f=  -4.68559004e- 1
x=  +4.00000002e- 1          f=  -9.53344002e- 1
x=  +1.30000001e+ 0          f=  -9.99270998e- 1
x=  +4.00000005e+ 0          f=  +7.28000092e+ 2
x=  -4.99999672e- 2          f=  +3.40095371e- 1
x=  +1.97500005e+ 0          f=  -1.40931413e- 1
x=  +9.62500058e- 1          f=  -1.00000000e+ 0
x=  -4.99999449e- 2          f=  +3.40095267e- 1
x=  +1.46875006e+ 0          f=  -9.89391647e- 1
x=  +7.09375061e- 1          f=  -9.99397442e- 1
x=  +1.08906257e+ 0          f=  -9.99999501e- 1
x=  +8.99213820e- 1          f=  -9.99998940e- 1
x=  +9.94140700e- 1          f=  -1.00000000e+ 0
```

ff[3]

```
x=  +9.99999996e- 2          f=  -6.51321560e- 1
x=  +4.00000002e- 1          f=  -9.93953384e- 1
x=  +1.30000001e+ 0          f=  -9.99994099e- 1
x=  +4.00000006e+ 0          f=  +5.90490122e+ 4
x=  -4.99999672e- 2          f=  +6.28894106e- 1
x=  +1.97500005e+ 0          f=  -2.23659944e- 1
x=  +9.62500058e- 1          f=  -1.00000000e+ 0
x=  -4.99999449e- 2          f=  +6.28893867e- 1
x=  +1.46875006e+ 0          f=  -9.99497832e- 1
x=  +7.09375061e- 1          f=  -9.99995701e- 1
x=  +1.08906257e+ 0          f=  -1.00000000e+ 0
```

ff[4]

```
x=  +9.99999996e- 2        f=  -8.61046580e- 2
x=  +4.00000002e- 1        f=  -2.20991079e- 1
x=  +1.30000001e+ 0        f=  -1.98258212e- 1
x=  -4.99999970e- 2        f=  +5.38993097e- 2
x=  +6.25000007e- 1        f=  -2.48756629e- 1
x=  +1.30000001e+ 0        f=  -1.98258212e- 1
x=  +2.87500005e- 1        f=  -1.67431701e- 1
x=  +7.93750010e- 1        f=  -2.47710001e- 1
x=  +5.40625006e- 1        f=  -2.43212853e- 1
x=  +6.67187504e- 1        f=  -2.49827083e- 1
x=  +7.93750005e- 1        f=  -2.47710003e- 1
x=  +5.03906251e- 1        f=  -2.47821726e- 1
x=  +6.95829131e- 1        f=  -2.49991976e- 1
x=  +7.93750010e- 1        f=  -2.47710001e- 1
x=  +5.51367202e- 1        f=  -2.49544926e- 1
x=  +7.22558610e- 1        f=  -2.49789994e- 1
x=  +6.86962903e- 1        f=  -2.49990381e- 1
x=  +7.04760753e- 1        f=  -2.49966670e- 1
x=  +6.95861831e- 1        f=  -2.49998163e- 1
x=  +6.95962993e- 1        f=  -2.49990331e- 1
x=  +7.00311296e- 1        f=  -2.49987261e- 1
x=  +6.93637105e- 1        f=  -2.49999940e- 1
x=  +6.96962910e- 1        f=  -2.49990378e- 1
x=  +6.95974203e- 1        f=  -2.49996351e- 1
x=  +6.91968553e- 1        f=  -2.49999654e- 1
x=  +6.94471374e- 1        f=  -2.49999560e- 1
x=  +6.93219967e- 1        f=  -2.49999998e- 1
x=  +6.91968553e- 1        f=  -2.49999654e- 1
x=  +6.93845674e- 1        f=  -2.49999877e- 1
x=  +6.92907117e- 1        f=  -2.49999935e- 1
```

ff[5]

```
x=  +1.09999999e+ 0        f=  +1.12211457e+ 0
x=  +9.49999996e- 1        f=  +1.04790844e+ 0
x=  +7.99999997e- 1        f=  +1.03414165e+ 0
x=  +3.49999994e- 1        f=  +1.00125054e+ 0
x=  -1.00000001e+ 0        f=  +1.08338295e+ 0
x=  +1.02500001e+ 0        f=  +1.09204436e+ 0
x=  +1.24999993e- 2        f=  +1.00000000e+ 0
x=  -1.00000000e+ 0        f=  +1.08338293e+ 0
x=  +5.16750004e- 1        f=  +1.00603490e+ 0
x=  -2.40624993e- 1        f=  +1.00027933e+ 0
x=  +1.39062503e- 1        f=  +1.00003117e+ 0
x=  -5.07812472e- 2        f=  +1.00000057e+ 0
x=  +4.41406281e- 2        f=  +1.00000033e+ 0
x=  -3.32030957e- 3        f=  +1.00000000e+ 0
```


The following codes are all multi-dimensional and within them are three more line search subroutines – the quadratic search of Powell, the cubic search of Davidon and a rational search.

(ii) The first of the multi-dimensional codes is that of Rosenbrock[5] and is based on a simple increase in search length for a successful step or a decrease in the opposite direction for an unsuccessful step. A solution of a set of linear algebraic equations solved as the minimum of a sum of squares is used as an example. Note the gentle convergence and being a sum of squares with zero minimum the full accuracy.

```
PROGRAM rosminnd
BEGIN
  INT n;
   n:=4;
  [1:n] REAL x;

   PROC rosmin=(PROC(REF[ ]REAL)REAL f,INT na,REF[ ]REAL x,
                INT iset,ifin,REAL ep,INT iwrit)VOID:
   BEGIN
      INT n:=na,n1,i;
      REAL f0,f1,b,gamma;
      [1:n] REAL x1,a,d,e,a1,prod;
      [1:n,1:n]REAL v,alpha,beta;
      [1:n]INT ii;

      PROC anorm=(REF[,]REAL beta,INT n1)REAL:
      BEGIN
         REAL s:=beta[n1,1]*beta[n1,1];
         FOR i1 FROM 2 TO n DO s PLUSAB beta[n1,i1]*beta[n1,i1] OD;
         sqrt(s)
      END;

      FORMAT for1=$l"x"10xn(n)(+d.7de+zd)l,"f"10x+d.7de+zd$,
             for2=$3l"gamma="+d.7de+zd2l,
                   n(n)("alpha("zd,")="+d.7de+zd)l,
                   "x"10xn(n)(+d.7de+zd)l,"f"10x+d.7de+zd$;
      clear(v);
      FOR i1 TO n DO v[i1,i1]:=1.0 OD;
      x1:=x;
      f0:=f(x);
      IF iwrit=3 THEN printf((for1,x,f0))FI;
      FOR icount WHILE
         IF icount=1 THEN TRUE
         ELIF iset=1 AND icount=ifin THEN FALSE
         ELIF iset/=1 AND
            (BOOL b1:=TRUE;
             FOR i1 TO n DO
                IF ABS(x1[i1]-x[i1])>ep THEN
                  b1:=FALSE;GOTO l1
                FI
            OD;
             l1:b1)
         THEN FALSE
         ELSE TRUE
         FI
      DO
         FOR i1 TO n DO
            a[i1]:=2.0;d[i1]:=0.0;e[i1]:=0.1
         OD;
         x1:=x;
         i:=0;
         WHILE
           (BOOL b1:=FALSE;
            i:=(i=n!1!i+1);
            FOR i1 TO n DO
               IF a[i1]>0.5 THEN
                  b1:=TRUE;
                  GOTO l1
               FI
            OD;
            l1:b1)
         DO
            FOR i1 TO n DO x[i1] PLUSAB e[i]*v[i,i1] OD;
            f1:=f(x);
            IF f1>=f0 THEN
               d[i] PLUSAB e[i];
               e[i] TIMESAB 3.0;
               IF iwrit/=1 THEN printf((for1,x,f1))FI;
               f0:=f1;
               IF a[i]>1.5 THEN a[i]:=1.0 FI
            ELSE
               FOR i1 TO n DO x[i1] MINUSAB e[i]*v[i,i1] OD;
               e[i] DIVAB (-2.0);
               IF a[i]<1.5 THEN a[i]:=0.0 FI
            FI
         OD;
         FOR i1 TO n DO
            FOR i2 TO n DO
               alpha[i1,i2]:=d[i1]*v[i1,i2];
               IF i1/=n THEN
                  FOR i3 FROM i1+1 TO n DO
                     alpha[i1,i2] PLUSAB d[i3]*v[i3,i2]
                  OD
               FI
            OD
         OD;
         beta[1, ]:=alpha[1, ];
         b:=anorm(beta,1);
         FOR i1 TO n DO v[1,i1]:=beta[1,i1]/b OD;
         FOR i1 FROM 2 TO n DO
            beta[i1, ]:=alpha[i1, ];
            FOR i2 TO i1-1 DO
               prod[i2]:=alpha[i1,1]*v[i2,1];
               FOR i3 FROM 2 TO n DO
                  prod[i2] PLUSAB alpha[i1,i3]*v[i2,i3]
               OD
            OD;
            FOR i2 TO i1-1 DO
               FOR i3 TO n DO
                  beta[i1,i3] MINUSAB prod[i2]*v[i2,i3]
               OD
            OD;
            b:=anorm(beta,i1);
            FOR i2 TO n DO v[i1,i2]:=beta[i1,i2]/b OD
         OD;
         FOR i1 TO n DO a1[i1]:=anorm(alpha,i1) OD;
         gamma:=a1[2]/a1[1];
         IF iwrit=3 THEN
            printf((for2,gamma));
            FOR i1 TO n DO printf((i1,a1[i1])) OD;
            printf((x,f1))
         FI
      OD;
      printf((for1,x,f1))
```

```
    END;

    PROC f=(REF[ ]REAL x)REAL:
    BEGIN
      REAL f1,f2,f3,f4;
      f1:=0.2317*x[1]+0.6123*x[2]+0.4137*x[3]+0.6696*x[4]-0.4735;
      f2:=0.4283*x[1]+0.8176*x[2]+0.4257*x[3]+0.3312*x[4]-0.2167;
       f3:=0.7321*x[1]+0.4135*x[2]+0.3126*x[3]+0.5163*x[4]-0.8152;
       f4:=0.8653*x[1]+0.2165*x[2]+0.8265*x[3]+0.7123*x[4]-0.1165;
      -(f1*f1+f2*f2+f3*f3+f4*f4)
    END;

    FOR i1 TO n DO  x[i1]:=0.0 OD;
    rosmin(f,n,x,1,30,1.0e-4,3)
END
FINISH

NT NEAR LINE 1
 Compatible with previous version of module
```



```
rosminnd

x          +0.0000000e+ 0+0.0000000e+ 0+0.0000000e+ 0+0.0000000e+ 0
f          -1.2009355e+ 0
x          +1.0000000e- 1+0.0000000e+ 0+0.0000000e+ 0+0.0000000e+ 0
f          -9.6711171e- 1
x          +1.0000000e- 1+1.0000000e- 1+0.0000000e+ 0+0.0000000e+ 0
f          -8.1611013e- 1
x          +1.0000000e- 1+1.0000000e- 1+1.0000000e- 1+0.0000000e+ 0
f          -6.7608245e- 1
x          +1.0000000e- 1+1.0000000e- 1+1.0000000e- 1+1.0000000e- 1
f          -5.2486119e- 1
x          +4.0000000e- 1+1.0000000e- 1+1.0000000e- 1+1.0000000e- 1
f          -2.2859882e- 1

gamma=+3.9735970e- 1

alpha( 1)=+4.3588990e- 1
alpha( 2)=+1.7320508e- 1
alpha( 3)=+1.4142136e- 1
alpha( 4)=+1.0000000e- 1

x          +4.0000001e- 1+1.0000000e- 1+1.0000000e- 1+1.0000000e- 1
f          -9.8339927e- 1
x          +4.9176630e- 1+1.2294158e- 1+1.2294158e- 1+1.2294158e- 1
f          -1.9120434e- 1
x          +4.9176630e- 1+1.2294157e- 1+5.2230897e- 2+1.9365225e- 1
f          -1.9069978e- 1
x          +5.1163428e- 1+9.6450925e- 2+2.5740249e- 2+1.6716161e- 1
f          -1.8442092e- 1
x          +5.1163428e- 1+1.3727575e- 1+5.3278350e- 3+1.4674919e- 1
f          -1.8363487e- 1

gamma=+7.7459667e- 1

alpha( 1)=+1.5811388e- 1
alpha( 2)=+1.2247449e- 1
alpha( 3)=+1.1180340e- 1
alpha( 4)=+1.0000000e- 1

x          +5.1163428e- 1+1.3727576e- 1+5.3278338e- 3+1.4674919e- 1
f          -1.8820607e- 1
x          +5.8223800e- 1+1.6085101e- 1-5.4548102e- 2+1.7631598e- 1
f          -1.7537587e- 1
x          +6.0037491e- 1+1.2921491e- 1-6.2093918e- 2+1.4295028e- 1
f          -1.6916548e- 1
x          +6.0037491e- 1+1.6572975e- 1-6.4539948e- 2+1.0888147e- 1
f          -1.6903621e- 1
x          +6.5332770e- 1+1.8341120e- 1-1.0944690e- 1+1.3105656e- 1
f          -1.6372312e- 1
x          +6.4952631e- 1+1.8276318e- 1-1.1435352e- 1+1.3071429e- 1
f          -1.6327421e- 1
x          +6.6312899e- 1+1.5903610e- 1-1.2001288e- 1+1.0569001e- 1
f          -1.6065918e- 1
```

```
gamma=+4.9976070e- 1

alpha( 1)=+2.0204037e- 1
alpha( 2)=+1.0097184e- 1
alpha( 3)=+1.0077822e- 1
alpha( 4)=+5.0000000e- 2

x             +6.6312900e- 1+1.5903610e- 1-1.2001289e- 1+1.0569001e- 1
f             -1.6349576e- 1
x             +7.3811140e- 1+1.6980639e- 1-1.8205035e- 1+8.5367744e- 2
f             -1.5264031e- 1
x             +9.6305858e- 1+2.0211727e- 1-3.6816273e- 1+2.4400939e- 2
f             -1.4190491e- 1
x             +9.5855936e- 1+2.2581710e- 1-3.6735272e- 1+1.7887948e- 2
f             -1.4146896e- 1
x             +9.5072750e- 1+2.2465826e- 1-3.7702698e- 1+1.7914585e- 2
f             -1.4138195e- 1
x             +9.5108122e- 1+2.2555909e- 1-3.7741100e- 1+2.0864071e- 2
f             -1.4129380e- 1
x             +9.4770680e- 1+2.4333396e- 1-3.7680349e- 1+1.5979328e- 2
f             -1.4117104e- 1
x             +9.8988440e- 1+2.4939225e- 1-4.1169956e- 1+4.5480522e- 3
f             -1.4105390e- 1


gamma=+9.9466737e- 2

alpha( 1)=+4.5852387e- 1
alpha( 2)=+4.5607874e- 2
alpha( 3)=+4.5500687e- 2
alpha( 4)=+4.3750001e- 2

x             +9.8989440e- 1+2.4939225e- 1-4.1169956e- 1+4.5480522e- 3
f             -1.4143829e- 1
x             +9.8097659e- 1+2.4692901e- 1-4.0374777e- 1+7.3053231e- 3
f             -1.4102684e- 1
x             +9.7841711e- 1+2.5249229e- 1-4.0454035e- 1+6.2833047e- 3
f             -1.4099965e- 1
x             +9.7756352e- 1+2.5108436e- 1-4.0385193e- 1+2.9117346e- 4
f             -1.4098658e- 1
x             +9.7842759e- 1+2.5162937e- 1-4.0267516e- 1+1.7614521e- 4
f             -1.4097123e- 1
x             +9.7650798e- 1+2.5579433e- 1-4.0326959e- 1-5.9034865e- 4
f             -1.4095454e- 1
```



```
gamma=+7.1252533e- 1

alpha( 1)=+1.7815241e- 2
alpha( 2)=+1.2693811e- 2
alpha( 3)=+6.4423510e- 3
alpha( 4)=+1.5625000e- 3

x             +9.7650798e- 1+2.5579433e- 1-4.0326959e- 1-5.9036864e- 4
f             -1.4098932e- 1
x             +9.7691543e- 1+2.5677938e- 1-4.0393653e- 1-1.5178712e- 3
f             -1.4095368e- 1
x             +9.7780545e- 1+2.5739288e- 1-4.0283674e- 1-1.2661402e- 3
f             -1.4095022e- 1
x             +9.7902777e- 1+2.6034805e- 1-4.0483754e- 1-4.0486491e- 3
f             -1.4094349e- 1
x             +9.7873447e- 1+2.6048843e- 1-4.0465270e- 1-4.1613164e- 3
f             -1.4094325e- 1
x             +9.7881620e- 1+2.6026772e- 1-4.0453012e- 1-4.4479644e- 3
f             -1.4094324e- 1


gamma=+9.9817351e- 1

alpha( 1)=+6.4659943e- 3
alpha( 2)=+6.4541843e- 3
alpha( 3)=+1.6105881e- 3
alpha( 4)=+1.5625000e- 3

x             +9.7881620e- 1+2.6026772e- 1-4.0453012e- 1-4.4479644e- 3
f             -1.4094565e- 1
x             +9.8104732e- 1+2.6459169e- 1-4.0574853e- 1-8.1766992e- 3
f             -1.4093775e- 1
x             +9.8272067e- 1+2.6783466e- 1-4.0664234e- 1-1.0973250e- 2
f             -1.4093712e- 1
x             +9.8211612e- 1+2.6808321e- 1-4.0628277e- 1-1.1170758e- 2
f             -1.4093655e- 1
x             +9.8216677e- 1+2.6758057e- 1-4.0617828e- 1-1.1757494e- 2
f             -1.4093626e- 1
x             +9.8236959e- 1+2.6761137e- 1-4.0584959e- 1-1.1707832e- 2
f             -1.4093406e- 1
x             +9.8297802e- 1+2.6770376e- 1-4.0486350e- 1-1.1558846e- 2
f             -1.4093302e- 1
x             +9.8252460e- 1+2.6789017e- 1-4.0457882e- 1-1.1706999e- 2
f             -1.4093179e- 1


gamma=+1.9877110e- 1

alpha( 1)=+1.1160191e- 2
alpha( 2)=+2.2183236e- 3
alpha( 3)=+1.7469281e- 3
alpha( 4)=+7.8125000e- 4

x             +9.8252460e- 1+2.6789017e- 1-4.0457882e- 1-1.1706999e- 2
f             -1.4097550e- 1
x             +9.8153976e- 1+2.6719399e- 1-3.9858357e- 1-1.2991879e- 2
f             -1.4093094e- 1
x             +9.7868520e- 1+2.6839527e- 1-3.9895256e- 1-1.3186289e- 2
f             -1.4092072e- 1
x             +9.7920440e- 1+2.6945246e- 1-3.9895948e- 1-1.4202602e- 2
```

```
f            -1.4091952e- 1
x            +9.8076200e- 1+2.7265404e- 1-3.9897993e- 1-1.7251539e- 2
f            -1.4091782e- 1
x            +9.8069395e- 1+2.7241548e- 1-3.9908020e- 1-1.7536131e- 2
f            -1.4091468e- 1
x.           +9.8050930e- 1+2.7228307e- 1-3.9795609e- 1-1.7777046e- 2
f            -1.4091298e- 1
x            +9.8030516e- 1+2.7156739e- 1-3.9825691e- 1-1.8630823e- 2
f            -1.4090949e- 1
x            +9.8147336e- 1+2.7396857e- 1-3.9827225e- 1-2.0917926e- 2
f            -1.4090593e- 1
x            +9.8091938e- 1+2.7357135e- 1-3.9489992e- 1-2.1640271e- 2
f            -1.4089911e- 1
x            +9.8038416e- 1+2.7379658e- 1-3.9496912e- 1-2.1676723e- 2
f            -1.4089877e- 1


gamma=+7.6650296e- 1

alpha( 1)=+1.5205552e- 2
alpha( 2)=+1.1655100e- 2
alpha( 3)=+4.0264704e- 3
alpha( 4)=+1.5625000e- 3

x            +9.8038416e- 1+2.7379658e- 1-3.9496912e- 1-2.1676723e- 2
f            -1.4093898e- 1
x            +9.6630740e- 1+3.1264038e- 1-3.3177055e- 1-8.7243066e- 2
f            -1.4077813e- 1
x            +9.7320054e- 1+3.1971036e- 1-3.3846086e- 1-9.0983206e- 2
f            -1.4073503e- 1
x            +9.7056125e- 1+3.2699357e- 1-3.2561113e- 1-1.0327690e- 1
f            -1.4071699e- 1
x            +9.6264308e- 1+3.4884321e- 1-2.9106194e- 1-1.4015796e- 1
f            -1.4069822e- 1
x            +9.6781300e- 1+3.5414569e- 1-2.9507967e- 1-1.4295807e- 1
f            -1.4065546e- 1
x            +9.6811602e- 1+3.5303315e- 1-2.9637770e- 1-1.4397451e- 1
f            -1.4064340e- 1
x            +9.8362578e- 1+3.6894060e- 1-3.1143089e- 1-1.5239882e- 1
f            -1.4064032e- 1
x            +9.8453483e- 1+3.6560297e- 1-3.1232500e- 1-1.5542614e- 1
f            -1.4056968e- 1
x            +9.7859620e- 1+3.8199020e- 1-2.8566311e- 1-1.8308494e- 1
f            -1.4051165e- 1
x            +9.7875226e- 1+3.8196108e- 1-2.8556017e- 1-1.8308649e- 1
f            -1.4050932e- 1
x            +9.7922045e- 1+3.8187371e- 1-2.8525138e- 1-1.8289112e- 1
f            -1.4050364e- 1
x            +9.8062502e- 1+3.8161161e- 1-2.8432498e- 1-1.8245501e- 1
f            -1.4049871e- 1
x            +9.7617105e- 1+3.9390203e- 1-2.6432856e- 1-2.0320861e- 1
f            -1.4046750e- 1
x            +9.7907913e- 1+3.9688468e- 1-2.6715103e- 1-2.0477848e- 1
f            -1.4045900e- 1


gamma=+2.1838197e- 1

alpha( 1)=+2.5498255e- 1
alpha( 2)=+5.5683591e- 2
```



```
alpha( 3)=+6.7460610e- 3
alpha( 4)=+6.2500000e- 3

x            +9.7907913e- 1+3.9688468e- 1-2.6715103e- 1-2.0477848e- 1
f            -1.4055796e- 1
x            +9.7856732e- 1+4.4515783e- 1-2.1702286e- 1-2.7658800e- 1
f            -1.4031700e- 1
x            +9.7703187e- 1+5.8997726e- 1-6.6638335e- 2-4.9201657e- 1
f            -1.3999404e- 1
x            +9.7242556e- 1+1.0244356e+ 0+3.8451524e- 1-1.1383023e+ 0
f            -1.3995238e- 1
x            +9.8796862e- 1+1.0339963e+ 0+3.6816737e- 1-1.1434652e+ 0
f            -1.3963300e- 1
x            +9.9822019e- 1+1.0144203e+ 0+3.7032541e- 1-1.1551243e+ 0
f            -1.3876816e- 1
x            +1.0448494e+ 0+1.0428024e+ 0+3.2128180e- 1-1.1706130e+ 0
f            -1.3810077e- 1
x            +1.0413946e+ 0+1.3686462e+ 0+6.5964698e- 1-1.6553273e+ 0
f            -1.3770234e- 1
x            +1.0490833e+ 0+1.3540392e+ 0+6.6126550e- 1-1.6640716e+ 0
f            -1.3715885e- 1
x            +1.0511686e+ 0+1.3544450e+ 0+6.6301567e- 1-1.6625920e+ 0
f            -1.3710659e- 1
x            +1.0861405e+ 0+1.3757316e+ 0+6.2623297e- 1-1.6742085e+ 0
f            -1.3683107e- 1


gamma=+8.1895889e- 2

alpha( 1)=+1.9816565e+ 0
alpha( 2)=+1.6228953e- 1
alpha( 3)=+4.3861465e- 2
alpha( 4)=+3.1250000e- 3

x            +1.0861405e+ 0+1.3757316e+ 0+6.2623297e- 1-1.6742085e+ 0
f            -1.3752202e- 1
x            +1.0915431e+ 0+1.4251269e+ 0+6.7131566e- 1-1.7483601e+ 0
f            -1.3665597e- 1
x            +1.1077510e+ 0+1.5733131e+ 0+8.0656371e- 1-1.9708149e+ 0
f            -1.3613519e- 1
x            +1.1563746e+ 0+2.0178715e+ 0+1.2123079e+ 0-2.6381792e+ 0
f            -1.3461417e- 1
x            +1.1743739e+ 0+2.0207908e+ 0+1.1964434e+ 0-2.6445684e+ 0
f            -1.3446314e- 1
x            +1.1811624e+ 0+1.9997531e+ 0+1.2038946e+ 0-2.6535577e+ 0
f            -1.3406307e- 1
x            +1.3270332e+ 0+3.3334284e+ 0+2.4211271e+ 0-4.6556509e+ 0
f            -1.2907983e- 1
x            +1.7646453e+ 0+7.3344545e+ 0+6.0728245e+ 0-1.0561930e+ 1
f            -1.1747402e- 1
x            +3.0774819e+ 0+1.9337533e+ 1+1.7027917e+ 1-2.8680768e+ 1
f            -1.1275187e- 1
x            +3.0909814e+ 0+1.9339722e+ 1+1.7016019e+ 1-2.8685550e+ 1
f            -1.1204153e- 1
x            +3.0960728e+ 0+1.9323944e+ 1+1.7021607e+ 1-2.8692302e+ 1
f            -1.0565428e- 1
x            +3.0940841e+ 0+1.9323366e+ 1+1.7019881e+ 1-2.8693882e+ 1
f            -1.0463948e- 1
x            +3.1345825e+ 0+1.9329934e+ 1+1.6984185e+ 1-2.8708257e+ 1
f            -1.0322346e- 1
```

```
x          +3.1498548e+ 0+1.9282599e+ 1+1.7000950e+ 1-2.8728483e+ 1
f          -8.9212702e- 2
x          +3.1438907e+ 0+1.9280665e+ 1+1.6995771e+ 1-2.8733222e+ 1
f          -8.8351789e- 2
x          +3.1897134e+ 0+1.9138860e+ 1+1.7046066e+ 1-2.8793900e+ 1
f          -8.6522164e- 2


gamma=+7.8850312e- 3

alpha( 1)=+3.6401132e+ 1
alpha( 2)=+2.8702406e- 1
alpha( 3)=+2.6904054e- 1
alpha( 4)=+1.2500000e- 2

x          +3.1897134e+ 0+1.9138860e+ 1+1.7046066e+ 1-2.8793900e+ 1
f          -1.0274538e- 1
x          +3.1954923e+ 0+1.9187658e+ 1+1.7091174e+ 1-2.8868403e+ 1
f          -8.6447514e- 2
x          +3.2128289e+ 0+1.9334053e+ 1+1.7226498e+ 1-2.9091910e+ 1
f          -8.6225783e- 2
x          +3.1889820e+ 0+1.9372012e+ 1+1.7224980e+ 1-2.9069816e+ 1
f          -8.0294491e- 2
x          +3.2213976e+ 0+1.9379298e+ 1+1.7253267e+ 1-2.9045403e+ 1
f          -7.7640335e- 2
x          +3.2734074e+ 0+1.9813483e+ 1+1.7659239e+ 1-2.9715924e+ 1
f          -7.6167982e- 2
x          +3.2586409e+ 0+1.9808350e+ 1+1.7676479e+ 1-2.9713268e+ 1
f          -7.5226437e- 2
x          +3.4146703e+ 0+2.1125904e+ 1+1.8894396e+ 1-3.1724831e+ 1
f          -7.1041335e- 2
x          +3.3703708e+ 0+2.1095505e+ 1+1.8946116e+ 1-3.1716864e+ 1
f          -6.9550086e- 2
x          +3.8384590e+ 0+2.5048166e+ 1+2.2599846e+ 1-3.7751553e+ 1
f          -5.9083872e- 2
x          +5.2427236e+ 0+3.6906149e+ 1+3.3561118e+ 1-5.5855620e+ 1
f          -4.1401047e- 2
x          +5.3091730e+ 0+3.6951748e+ 1+3.3483537e+ 1-5.5867571e+ 1
f          -4.0891293e- 2


gamma=+1.9727168e- 3

alpha( 1)=+3.6400072e+ 1
alpha( 2)=+7.1807032e- 2
alpha( 3)=+5.1538820e- 2
alpha( 4)=+5.0000000e- 2

x          +5.3091729e+ 0+3.6951748e+ 1+3.3483537e+ 1-5.5867571e+ 1
f          -1.0365559e- 1
x          +5.3149956e+ 0+3.7000685e+ 1+3.3528695e+ 1-5.5941949e+ 1
f          -4.0799192e- 2
x          +5.2734170e+ 0+3.6964898e+ 1+3.3609317e+ 1-5.5919801e+ 1
f          -3.4890654e- 2
x          +5.2908850e+ 0+3.7111707e+ 1+3.3744791e+ 1-5.6142935e+ 1
f          -3.4615179e- 2
x          +5.3432891e+ 0+3.7552135e+ 1+3.4151211e+ 1-5.6812337e+ 1
f          -3.3798642e- 2
x          +5.3488401e+ 0+3.7569627e+ 1+3.4157507e+ 1-5.6796571e+ 1
f          -3.3006094e- 2
```



```
x          +5.5060524e+ 0+3.8890910e+ 1+3.5376768e+ 1-5.8804779e+ 1
f          -3.0953075e- 2
x          +5.4950522e+ 0+3.8895635e+ 1+3.5373174e+ 1-5.8904713e+ 1
f          -3.0447693e- 2
x          +5.4638683e+ 0+3.8869794e+ 1+3.5433640e+ 1-5.8788102e+ 1
f          -2.9939723e- 2
x          +5.9355053e+ 0+4.2832643e+ 1+3.9091422e+ 1-6.4812726e+ 1
f          -2.4365793e- 2
x          +5.9271790e+ 0+4.2806405e+ 1+3.9081978e+ 1-6.4836375e+ 1
f          -2.4298376e- 2
x          +7.3420901e+ 0+5.4697952e+ 1+5.0055324e+ 1-8.2910247e+ 1
f          -1.0628941e- 2


gamma=+4.8321026e- 3

alpha( 1)=+3.6400425e+ 1
alpha( 2)=+1.7589059e- 1
alpha( 3)=+1.7544586e- 1
alpha( 4)=+1.7500000e- 1

x          +7.3420901e+ 0+5.4697952e+ 1+5.0055323e+ 1-8.2910246e+ 1
f          -3.4423046e- 2
x          +7.3476750e+ 0+5.4745705e+ 1+5.0100850e+ 1-8.2984540e+ 1
f          -1.0579704e- 2
x          +7.2984474e+ 0+5.4708576e+ 1+5.0177011e+ 1-8.2966589e+ 1
f          -8.4997731e- 3
x          +7.3152021e+ 0+5.4854835e+ 1+5.0313591e+ 1-8.3189466e+ 1
f          -8.3698774e- 3
x          +7.3654659e+ 0+5.5293609e+ 1+5.0723378e+ 1-8.3858095e+ 1
f          -7.9900862e- 3
x          +7.5162574e+ 0+5.6609933e+ 1+5.1952539e+ 1-8.5863984e+ 1
f          -6.9408574e- 3
x          +7.5056554e+ 0+5.6614964e+ 1+5.1948237e+ 1-8.5964116e+ 1
f          -6.5731839e- 3
x          +7.9580390e+ 0+6.0563936e+ 1+5.5635872e+ 1-9.1581783e+ 1
f          -3.9133008e- 3
x          +9.3151540e+ 0+7.2410851e+ 1+6.6598776e+ 1-1.0993479e+ 2
f          -3.2562268e- 3
x          +9.3145679e+ 0+7.2408744e+ 1+6.6697817e+ 1-1.0993680e+ 2
f          -2.8911250e- 3
x          +9.3191831e+ 0+7.2412319e+ 1+6.6590677e+ 1-1.0993849e+ 2
f          -2.8448856e- 3
x          +9.3174249e+ 0+7.2405996e+ 1+6.6687799e+ 1-1.0994453e+ 2
f          -2.2951497e- 3
x          +9.3094734e+ 0+7.2409770e+ 1+6.6684572e+ 1-1.0994463e+ 2
f          -1.7379470e- 3


gamma=+2.5841119e- 3

alpha( 1)=+3.6400123e+ 1
alpha( 2)=+9.4061991e- 2
alpha( 3)=+2.5194579e- 2
alpha( 4)=+2.1875000e- 2

x          +9.3094734e+ 0+7.2409770e+ 1+6.6684572e+ 1-1.0994463e+ 2
f          -5.7479871e- 3
x          +9.3067709e+ 0+7.2385441e+ 1+6.6661730e+ 1-1.0990749e+ 2
f          -1.7345917e- 3
```

```
x            +9.2986636e+ 0+7.2312453e+ 1+6.6593203e+ 1-1.0979609e+ 2
f            -1.7246912e- 3
x            +9.2812486e+ 0+7.2303338e+ 1+6.6508497e+ 1-1.0979392e+ 2
f            -1.6514653e- 3
x            +9.2747993e+ 0+7.2322971e+ 1+6.6510889e+ 1-1.0978006e+ 2
f            -1.3147616e- 3
x            +9.2504774e+ 0+7.2104008e+ 1+6.6405309e+ 1-1.0944584e+ 2
f            -1.2786563e- 3
x            +9.1982324e+ 0+7.2076661e+ 1+6.6451192e+ 1-1.0943933e+ 2
f            -9.0649223e- 4
x            +9.1252866e+ 0+7.1419769e+ 1+6.5934450e+ 1-1.0843669e+ 2
f            -7.7733961e- 4
x            +8.9063889e+ 0+6.9449093e+ 1+6.3984223e+ 1-1.0542875e+ 2
f            -5.7722387e- 4
x            +8.9084542e+ 0+6.9448729e+ 1+6.3986219e+ 1-1.0542761e+ 2
f            -5.5598466e- 4
```

```
gamma=+1.7042979e- 2

alpha( 1)=+6.0508789e+ 0
alpha( 2)=+1.0312500e- 1
alpha( 3)=+2.5194554e- 2
alpha( 4)=+2.5000000e- 2
```

```
x            +8.9084542e+ 0+6.9448729e+ 1+6.3986219e+ 1-1.0542761e+ 2
f            -8.9286145e- 4
x            +8.9018267e+ 0+6.9399793e+ 1+6.3941625e+ 1-1.0535296e+ 2
f            -5.5394111e- 4
x            +8.8819444e+ 0+6.9252985e+ 1+6.3807842e+ 1-1.0512901e+ 2
f            -5.5119534e- 4
x            +8.8819534e+ 0+6.9258177e+ 1+6.3807964e+ 1-1.0512553e+ 2
f            -4.4540996e- 4
x            +8.8776200e+ 0+6.9259508e+ 1+6.3804078e+ 1-1.0512737e+ 2
f            -3.2975873e- 4
x            +8.8850759e+ 0+6.9314561e+ 1+6.3854246e+ 1-1.0521135e+ 2
f            -3.2737538e- 4
x            +8.8873159e+ 0+6.9315056e+ 1+6.3852235e+ 1-1.0521203e+ 2
f            -3.2317738e- 4
x            +8.8940357e+ 0+6.9316541e+ 1+6.3846203e+ 1-1.0521406e+ 2
f            -3.2226717e- 4
x            +8.8828518e+ 0+6.9233962e+ 1+6.3770950e+ 1-1.0508809e+ 2
f            -3.2000555e- 4
```

```
gamma=+3.3535780e- 2

alpha( 1)=+4.5650679e- 1
alpha( 2)=+1.5309311e- 2
alpha( 3)=+8.8388348e- 3
alpha( 4)=+6.2500000e- 3
```

```
x            +8.8828518e+ 0+6.9233962e+ 1+6.3770950e+ 1-1.0508809e+ 2
f            -4.0934118e- 4
x            +8.8772435e+ 0+6.9185916e+ 1+6.3723794e+ 1-1.0501371e+ 2
f            -2.7885009e- 4
x            +8.8604186e+ 0+6.9045779e+ 1+6.3582326e+ 1-1.0479059e+ 2
f            -1.7806738e- 4
x            +8.8099436e+ 0+6.8622369e+ 1+6.3157922e+ 1-1.0412122e+ 2
f            -7.9811301e- 5
```



```
x            +8.8089910e+ 0+6.8620583e+ 1+6.3160284e+ 1-1.0412092e+ 2
f            -7.8901420e- 5
x            +8.8105297e+ 0+6.8621947e+ 1+6.3161701e+ 1-1.0411904e+ 2
f            -7.2571304e- 5
x            +8.8092589e+ 0+6.8622746e+ 1+6.3161738e+ 1-1.0411861e+ 2
f            -7.1393205e- 5
```

```
gamma=+3.6057457e- 3

alpha( 1)=+1.3000085e+ 0
alpha( 2)=+4.6875000e- 3
alpha( 3)=+3.4938562e- 3
alpha( 4)=+3.1250000e- 3
```

```
x            +8.8092589e+ 0+6.8622745e+ 1+6.3161738e+ 1-1.0411861e+ 2
f            -7.9430523e- 5
x            +8.8120894e+ 0+6.8646254e+ 1+6.3185170e+ 1-1.0415590e+ 2
f            -6.7690949e- 5
x            +8.8205808e+ 0+6.8716778e+ 1+6.3255463e+ 1-1.0426776e+ 2
f            -6.2953080e- 5
x            +8.8176194e+ 0+6.8717596e+ 1+6.3254896e+ 1-1.0426783e+ 2
f            -5.5347113e- 5
x            +8.8180543e+ 0+6.8719906e+ 1+6.3254449e+ 1-1.0426725e+ 2
f            -5.4099140e- 5
x            +8.8196455e+ 0+6.8732130e+ 1+6.3267629e+ 1-1.0428822e+ 2
f            -5.2977650e- 5
x            +8.8197604e+ 0+6.8732066e+ 1+6.3266791e+ 1-1.0428865e+ 2
f            -5.2413925e- 5
x            +8.8245369e+ 0+6.8771736e+ 1+6.3306531e+ 1-1.0435158e+ 2
f            -5.2066947e- 5
```

```
gamma=+1.1455687e- 2

alpha( 1)=+3.1252051e- 1
alpha( 2)=+3.5801373e- 3
alpha( 3)=+3.4938562e- 3
alpha( 4)=+3.1250000e- 3
```

```
x            +8.8245369e+ 0+6.8771736e+ 1+6.3306531e+ 1-1.0435158e+ 2
f            -7.0315784e- 5
x            +8.8239259e+ 0+6.8765777e+ 1+6.3300740e+ 1-1.0434226e+ 2
f            -5.2038422e- 5
x            +8.8244514e+ 0+6.8766272e+ 1+6.3300603e+ 1-1.0434199e+ 2
f            -5.1579510e- 5
x            +8.8241880e+ 0+6.8765495e+ 1+6.3300421e+ 1-1.0434198e+ 2
f            -5.0727592e- 5
x            +8.8242453e+ 0+6.8767054e+ 1+6.3300964e+ 1-1.0434285e+ 2
f            -5.0677224e- 5
x            +8.8234550e+ 0+6.8767723e+ 1+6.3300417e+ 1-1.0434282e+ 2
f            -4.9658138e- 5
x            +8.8236269e+ 0+6.8769399e+ 1+6.3302046e+ 1-1.0434544e+ 2
f            -4.9585956e- 5
x            +8.8235976e+ 0+6.8769420e+ 1+6.3302118e+ 1-1.0434538e+ 2
f            -4.9495369e- 5
x            +8.8241131e+ 0+6.8774447e+ 1+6.3307004e+ 1-1.0435324e+ 2
f            -4.9265982e- 5
x            +8.8240252e+ 0+6.8774510e+ 1+6.3307219e+ 1-1.0435308e+ 2
f            -4.9131566e- 5
```

```
x          +8.8255719e+ 0+6.8789595e+ 1+6.3321878e+ 1-1.0437666e+ 2
f          -4.8396160e- 5


gamma=+5.2004259e- 2

alpha( 1)=+3.4421578e- 2
alpha( 2)=+1.7900686e- 3
alpha( 3)=+8.7346405e- 4
alpha( 4)=+7.8124999e- 4

x          +8.8255719e+ 0+6.8789595e+ 1+6.3321878e+ 1-1.0437666e+ 2
f          -4.9122373e- 5
x          +8.8263235e+ 0+6.8802565e+ 1+6.3333025e+ 1-1.0439488e+ 2
f          -4.7279147e- 5
x          +8.8264644e+ 0+6.8804996e+ 1+6.3335114e+ 1-1.0439830e+ 2
f          -4.7218426e- 5
x          +8.8266869e+ 0+6.8805033e+ 1+6.3334828e+ 1-1.0439844e+ 2
f          -4.7094190e- 5
x          +8.8267577e+ 0+6.8804877e+ 1+6.3334896e+ 1-1.0439850e+ 2
f          -4.7074112e- 5
x          +8.8268634e+ 0+6.8805702e+ 1+6.3336463e+ 1-1.0440107e+ 2
f          -4.6994582e- 5
x          +8.8269354e+ 0+6.8805731e+ 1+6.3336500e+ 1-1.0440102e+ 2
f          -4.6910937e- 5
x          +8.8272525e+ 0+6.8812203e+ 1+6.3341202e+ 1-1.0440871e+ 2
f          -4.6878430e- 5


gamma=+1.0228429e- 2

alpha( 1)=+4.3752289e- 2
alpha( 2)=+4.4751716e- 4
alpha( 3)=+4.0264703e- 4
alpha( 4)=+3.9062500e- 4

x          +8.8272525e+ 0+6.8812203e+ 1+6.3341202e+ 1-1.0440871e+ 2
f          -4.7338559e- 5
x          +8.8273125e+ 0+6.8813011e+ 1+6.3341892e+ 1-1.0440985e+ 2
f          -4.6874205e- 5
x          +8.8285871e+ 0+6.8812688e+ 1+6.3341250e+ 1-1.0441040e+ 2
f          -4.6111562e- 5
x          +8.8287671e+ 0+6.8815110e+ 1+6.3343320e+ 1-1.0441383e+ 2
f          -4.5956932e- 5
x          +8.8293072e+ 0+6.8822376e+ 1+6.3349531e+ 1-1.0442413e+ 2
f          -4.5726184e- 5
x          +8.8302631e+ 0+6.8822134e+ 1+6.3349050e+ 1-1.0442454e+ 2
f          -4.5422517e- 5
x          +8.8306692e+ 0+6.8827584e+ 1+6.3353708e+ 1-1.0443227e+ 2
f          -4.5153813e- 5
x          +8.8306686e+ 0+6.8827546e+ 1+6.3353736e+ 1-1.0443228e+ 2
f          -4.5142036e- 5
x          +8.8306967e+ 0+6.8827559e+ 1+6.3353763e+ 1-1.0443225e+ 2
f          -4.5140240e- 5
x          +8.8314136e+ 0+6.8827378e+ 1+6.3353402e+ 1-1.0443256e+ 2
f          -4.5117990e- 5


gamma=+1.1631509e- 1
```



```
alpha( 1)=+3.1070269e- 2
alpha( 2)=+3.6139411e- 3
alpha( 3)=+6.9053400e- 5
alpha( 4)=+4.8828125e- 5

x          +8.8314136e+ 0+6.8827378e+ 1+6.3353402e+ 1-1.0443256e+ 2
f          -4.5176187e- 5
x          +8.8322507e+ 0+6.8830431e+ 1+6.3355856e+ 1-1.0443736e+ 2
f          -4.5085673e- 5
x          +8.8316152e+ 0+6.8830644e+ 1+6.3356204e+ 1-1.0443715e+ 2
f          -4.4913505e- 5
x          +8.8317722e+ 0+6.8831216e+ 1+6.3356664e+ 1-1.0443805e+ 2
f          -4.4895371e- 5
x          +8.8322431e+ 0+6.8832932e+ 1+6.3358044e+ 1-1.0444075e+ 2
f          -4.4842375e- 5
x          +8.8317665e+ 0+6.8833092e+ 1+6.3358305e+ 1-1.0444060e+ 2
f          -4.4814692e- 5
x          +8.8318053e+ 0+6.8833056e+ 1+6.3358383e+ 1-1.0444058e+ 2
f          -4.4809439e- 5
x          +8.8332180e+ 0+6.8839207e+ 1+6.3362525e+ 1-1.0444867e+ 2
f          -4.4642136e- 5
x          +8.8331983e+ 0+6.8839171e+ 1+6.3362526e+ 1-1.0444870e+ 2
f          -4.4628111e- 5


gamma=+6.3709274e- 2

alpha( 1)=+2.1528109e- 2
alpha( 2)=+1.3715402e- 3
alpha( 3)=+1.0918301e- 4
alpha( 4)=+4.8828126e- 5

x          +8.8331983e+ 0+6.8839171e+ 1+6.3362526e+ 1-1.0444870e+ 2
f          -4.4665373e- 5
x          +8.8352706e+ 0+6.8850704e+ 1+6.3373121e+ 1-1.0446744e+ 2
f          -4.4274284e- 5
x          +8.8368249e+ 0+6.8860105e+ 1+6.3381067e+ 1-1.0448150e+ 2
f          -4.4176053e- 5
x          +8.8355659e+ 0+6.8860397e+ 1+6.3381809e+ 1-1.0448103e+ 2
f          -4.3976444e- 5
x          +8.8359432e+ 0+6.8859339e+ 1+6.3382895e+ 1-1.0448108e+ 2
f          -4.3931437e- 5
x          +8.8371090e+ 0+6.8865389e+ 1+6.3388854e+ 1-1.0449162e+ 2
f          -4.3799198e- 5
x          +8.8368987e+ 0+6.8865192e+ 1+6.3388723e+ 1-1.0449185e+ 2
f          -4.2840659e- 5
x          +8.8403958e+ 0+6.8887342e+ 1+6.3406601e+ 1-1.0452348e+ 2
f          -4.2526987e- 5
x          +8.8394538e+ 0+6.8887562e+ 1+6.3407158e+ 1-1.0452313e+ 2
f          -4.1990622e- 5


gamma=+3.1718557e- 2

alpha( 1)=+1.0005034e- 1
alpha( 2)=+3.1734525e- 3
alpha( 3)=+1.6105881e- 3
alpha( 4)=+3.9062500e- 4

x          +8.8394538e+ 0+6.8887562e+ 1+6.3407158e+ 1-1.0452313e+ 2
```

```
f            -4.6398791e- 5
x            +8.8457062e+ 0+6.8936927e+ 1+6.3451768e+ 1-1.0459752e+ 2
f            -4.0193037e- 5
x            +8.8644634e+ 0+6.9085023e+ 1+6.3585597e+ 1-1.0482069e+ 2
f            -3.9836954e- 5
x            +8.8604435e+ 0+6.9083461e+ 1+6.3589948e+ 1-1.0481945e+ 2
f            -3.4666453e- 5
x            +8.8612180e+ 0+6.9082262e+ 1+6.3590369e+ 1-1.0481993e+ 2
f            -3.1629225e- 5
x            +8.8620973e+ 0+6.9089204e+ 1+6.3596642e+ 1-1.0483089e+ 2
f            -3.1503263e- 5
x            +8.8618706e+ 0+6.9089077e+ 1+6.3596449e+ 1-1.0483061e+ 2
f            -2.9124242e- 5
x            +8.8645084e+ 0+6.9109903e+ 1+6.3615269e+ 1-1.0486189e+ 2
f            -2.8491541e- 5
x            +8.8637546e+ 0+6.9109611e+ 1+6.3616085e+ 1-1.0486176e+ 2
f            -2.8168998e- 5
x            +8.8643355e+ 0+6.9108712e+ 1+6.3616401e+ 1-1.0486212e+ 2
f            -2.8047581e- 5


gamma=+1.7354522e- 2

alpha( 1)=+4.5631872e- 1
alpha( 2)=+7.9191934e- 3
alpha( 3)=+2.7621359e- 3
alpha( 4)=+3.9062500e- 4

x            +8.8643355e+ 0+6.9108712e+ 1+6.3616401e+ 1-1.0486212e+ 2
f            -3.1426155e- 5
x            +8.8697892e+ 0+6.9157176e+ 1+6.3662255e+ 1-1.0493641e+ 2
f            -2.5757938e- 5
x            +8.8861462e+ 0+6.9302568e+ 1+6.3799819e+ 1-1.0515928e+ 2
f            -2.0516298e- 5
x            +8.9352202e+ 0+6.9739745e+ 1+6.4212508e+ 1-1.0582748e+ 2
f            -1.9559568e- 5
x            +8.9366617e+ 0+6.9740381e+ 1+6.4210279e+ 1-1.0582808e+ 2
f            -1.5402725e- 5
x            +8.9182589e+ 0+6.9576815e+ 1+6.4055521e+ 1-1.0557786e+ 2
f            -1.4811371e- 5
x            +8.9172519e+ 0+6.9575482e+ 1+6.4054698e+ 1-1.0557806e+ 2
f            -1.3165093e- 5
x            +8.9167775e+ 0+6.9577003e+ 1+6.4054744e+ 1-1.0557792e+ 2
f            -1.0353460e- 5
x            +8.9443817e+ 0+6.9822352e+ 1+6.4286881e+ 1-1.0595391e+ 2
f            -9.7611263e- 6
x            +8.9454628e+ 0+6.9823580e+ 1+6.4285210e+ 1-1.0595406e+ 2
f            -5.7029682e- 6
x            +8.9440393e+ 0+6.9825144e+ 1+6.4285347e+ 1-1.0595306e+ 2
f            -3.6927473e- 6


gamma=+4.4183738e- 3

alpha( 1)=+1.4687643e+ 0
alpha( 2)=+6.4895499e- 3
alpha( 3)=+3.4938562e- 3
alpha( 4)=+1.5625000e- 3

x            +8.9440393e+ 0+6.9825144e+ 1+6.4285347e+ 1-1.0595306e+ 2
```



```
f            -4.9222130e- 5
x            +8.9494660e+ 0+6.9873921e+ 1+6.4330892e+ 1-1.0602734e+ 2
f            -3.2153015e- 6
x            +8.9657458e+ 0+7.0020255e+ 1+6.4467525e+ 1-1.0625017e+ 2
f            -2.6781456e- 6
x            +8.9656534e+ 0+7.0021363e+ 1+6.4466429e+ 1-1.0625012e+ 2
f            -2.2161322e- 6
x            +8.9641272e+ 0+7.0007645e+ 1+6.4453619e+ 1-1.0622923e+ 2
f            -2.2110661e- 6
x            +8.9637564e+ 0+7.0007700e+ 1+6.4453709e+ 1-1.0622916e+ 2
f            -2.1609780e- 6
x            +8.9636374e+ 0+7.0007509e+ 1+6.4453514e+ 1-1.0622942e+ 2
f            -1.2272292e- 6
x            +8.9659266e+ 0+7.0028088e+ 1+6.4472729e+ 1-1.0626075e+ 2
f            -1.2246481e- 6


gamma=+4.0024591e- 3

alpha( 1)=+4.1406582e- 1
alpha( 2)=+1.6572815e- 3
alpha( 3)=+5.5242717e- 4
alpha( 4)=+3.9062500e- 4

x            +8.9659266e+ 0+7.0028088e+ 1+6.4472729e+ 1-1.0626075e+ 2
f            -7.5898559e- 6
x            +8.9632837e+ 0+7.0003581e+ 1+6.4450102e+ 1-1.0622360e+ 2
f            -1.1749757e- 6
x            +8.9634616e+ 0+7.0003408e+ 1+6.4449901e+ 1-1.0622382e+ 2
f            -9.8040410e- 7
x            +8.9635303e+ 0+7.0003294e+ 1+6.4450043e+ 1-1.0622381e+ 2
f            -8.8929676e- 7
x            +8.9636897e+ 0+7.0003386e+ 1+6.4450032e+ 1-1.0622374e+ 2
f            -8.3144011e- 7


gamma=+9.5678814e- 3

alpha( 1)=+5.0002288e- 2
alpha( 2)=+4.7841596e- 4
alpha( 3)=+4.3673202e- 4
alpha( 4)=+3.9062500e- 4

x            +8.9636897e+ 0+7.0003386e+ 1+6.4450032e+ 1-1.0622374e+ 2
f            -2.1825078e- 6
x            +8.9625713e+ 0+6.9991035e+ 1+6.4438684e+ 1-1.0620524e+ 2
f            -7.8007583e- 7
x            +8.9629029e+ 0+6.9990876e+ 1+6.4438629e+ 1-1.0620536e+ 2
f            -7.0649262e- 7
x            +8.9629008e+ 0+6.9991055e+ 1+6.4438293e+ 1-1.0620544e+ 2
f            -5.1699812e- 7
x            +8.9629270e+ 0+6.9991344e+ 1+6.4438559e+ 1-1.0620588e+ 2
f            -5.1615325e- 7
x            +8.9631758e+ 0+6.9991225e+ 1+6.4438518e+ 1-1.0620597e+ 2
f            -4.8898458e- 7
x            +8.9631741e+ 0+6.9991359e+ 1+6.4438265e+ 1-1.0620603e+ 2
f            -4.6598878e- 7
x            +8.9631999e+ 0+6.9991389e+ 1+6.4438273e+ 1-1.0620601e+ 2
f            -4.1752581e- 7
x            +8.9632770e+ 0+6.9991480e+ 1+6.4438300e+ 1-1.0620592e+ 2
```

```
f          -4.0974664e- 7
x          +8.9632967e+ 0+6.9991697e+ 1+6.4438499e+ 1-1.0620625e+ 2
f          -4.0854804e- 7

gamma=+4.1103745e- 2

alpha( 1)=+2.3994888e- 2
alpha( 2)=+9.8627977e- 4
alpha( 3)=+7.1094822e- 4
alpha( 4)=+1.9531250e- 4

x          +8.9632967e+ 0+6.9991697e+ 1+6.4438499e+ 1-1.0620625e+ 2
f          -1.6086580e- 6
x          +8.9628870e+ 0+6.9979519e+ 1+6.4425484e+ 1-1.0618803e+ 2
f          -3.9347417e- 7
x          +8.9630406e+ 0+6.9984086e+ 1+6.4430990e+ 1-1.0619486e+ 2
f          -3.4875631e- 7
x          +8.9633101e+ 0+6.9984156e+ 1+6.4430740e+ 1-1.0619497e+ 2
f          -3.2320322e- 7
x          +8.9631088e+ 0+6.9984435e+ 1+6.4430570e+ 1-1.0619490e+ 2
f          -2.5096597e- 7
x          +8.9633110e+ 0+6.9984488e+ 1+6.4430392e+ 1-1.0619499e+ 2
f          -2.4132743e- 7
x          +8.9632862e+ 0+6.9984465e+ 1+6.4430362e+ 1-1.0619501e+ 2
f          -2.3641267e- 7

gamma=+5.0421704e- 2

alpha( 1)=+1.5644900e- 2
alpha( 2)=+7.8884251e- 4
alpha( 3)=+3.9366493e- 4
alpha( 4)=+4.8828125e- 5

x          +8.9632862e+ 0+6.9984465e+ 1+6.4430362e+ 1-1.0619501e+ 2
f          -3.7350841e- 7
x          +8.9632862e+ 0+6.9984465e+ 1+6.4430362e+ 1-1.0619501e+ 2
f          -3.7350841e- 7
```


(iv) **The second multiple method is again a direct search and is the simplex method of Nelder and Mead[6].**

```
PROGRAM simplex
 BEGIN
    INT n;
   n:=2;
    FORMAT fq1=$l4(4(3x+d.8de+zd)l)$,
           fq2=$l+d.8de+zd$;

    PROC f=( REF [ ] REAL x) REAL :
    BEGIN
       REAL f1,f2,w1,w2,w3;
      w3:=x[1]-x[2];
       IF  ABS w3<=1.0e-20 THEN
         w1:=cos(x[2]);
         w2:=sin(x[2]);
         f1:=x[2]*x[2]*(-w2-w3*w1/2)/2+x[2]*(w1+w3*x[2]*w1/2+x[2]*w2)-w2;
         f2:=(x[2]*x[2]-pi*pi/4)*(-w2-w3*w1/2)/2
             +(x[2]-pi/2)*(w1+w3*x[2]*w1/2+x[2]*w2)+1-w2
       ELSE
      w1:=cos(x[1]);
      w2:=cos(x[2]);
      f1:=x[2]*x[2]*(w1-w2)/w3/2.0+x[2]*(x[1]*w2-x[2]*w1)/w3-sin(x[2]);
      f2:=(x[1]*x[1]-pi*pi/4.0)*(w1-w2)/w3/2.0
         +(x[1]-pi/2.0)*(x[1]*w2-x[2]*w1)/w3+1.0-sin(x[1])
       FI ;
      f1*f1+f2*f2
     END ;

   [1:n+1,1:n] REAL p;
    REAL e;

    PROC simplex=( INT nn, PROC ( REF [ ] REAL ) REAL f,
                   REF [,] REAL p, REAL e) VOID :
    BEGIN
       INT n:=nn+1;
      [1:n] REAL y;
      [1:n] REAL yrem;
       REAL ys,yss,ma,mi,s1,yh,yl;
      [1:nn] REAL ps,pss,pbar;
       REAL alpha,beta,gamma;
       INT h,l,ul;
       BOOL bql:= FALSE ;
       INT ino;
       FORMAT fur1=$l6(+d.8de+zd3x)$;
      alpha:=1.0;
      beta:=0.5;
      gamma:=2.0;
      ul:=250;
       FOR i1 TO n DO y[i1]:=f(p[i1, ]) OD ;
       FOR ic TO ul
       WHILE
          IF ic=1 THEN  TRUE  ELSE
             BOOL b:= FALSE ;
             FOR i1 TO n DO
            ( ABS y[i1]>e*e*0.01!bql:= FALSE ; GOTO l2) OD ;
```



```
            (bql!ino PLUSAB 1!ino:=1;bql:= TRUE );
            (ino=4!b:= FALSE ; GOTO l1);
         l2: FOR i1 TO n DO
             FOR i1 TO n DO
            ( ABS (yrem[i1]-y[i1])>=e* ABS y[i1]!b:= TRUE ; GOTO l1) OD  OD ;
         l1:b
          FI
      DO
      BEGIN
         ma:=y[1];
         mi:=ma;
         h:=l:=1;
       yrem:=y;
          FOR i1 FROM 2 TO n DO
         ( IF y[i1]>ma THEN ma:=y[i1];h:=i1 FI ;
           IF y[i1]<mi THEN mi:=y[i1];l:=i1 FI ) OD ;
          FOR i1 TO nn DO
         (s1:=0.0;
           FOR i2 TO n DO (i2/=h!s1 PLUSAB p[i2,i1]) OD ;
         pbar[i1]:=s1/(n-1)) OD ;
          FOR i1 TO nn DO ps[i1]:=(1.0+alpha)*pbar[i1]-alpha*p[h,i1] OD ;
         ys:=f(ps);yl:=y[l];yh:=y[h];
          IF ys<yl THEN
             FOR i1 TO nn DO pss[i1]:=gamma*ps[i1]+(1.0-gamma)*pbar[i1] OD ;
            yss:=f(pss);
             IF yss<yl THEN p[h, ]:=pss;y[h]:=yss ELSE p[h, ]:=ps;y[h]:=ys FI
          ELIF
             BOOL b:= FALSE ;
             FOR i1 TO n WHILE NOT b DO (i1/=h!(y[i1]>ys!b:= TRUE )) OD;
             b
          THEN
            p[h, ]:=ps;y[h]:=ys
          ELSE
             IF ys<=yh THEN p[h, ]:=ps;yh:=ys FI ;
             FOR i1 TO nn DO pss[i1]:=beta*p[h,i1]+(1.0-beta)*pbar[i1] OD ;
            yss:=f(pss);
             IF yss>yh THEN
                FOR i1 TO nn DO
                FOR i2 TO n DO p[i2,i1]:=(p[i2,i1]+p[l,i1])/2.0 OD  OD ;
                FOR i1 TO n DO y[i1]:=f(p[i1, ]) OD
             ELSE p[h, ]:=pss;y[h]:=yss
             FI
          FI
         ;printf((fur1,y));
         printf((fur1,p[l, ]))
      END  OD ;
     print((newline,"min f",newline));
     printf((fq2,y[l]));
     print((newline,"min point",newline));
     printf((fur1,p[l, ]))
   END ;

   FOR i1 TO n+1 DO  FOR i2 TO n DO
     read(p[i1,i2]) OD  OD ;
  simplex(n,f,p,1.0e-08)
 END
 FINISH
```

```
simplex
0.3    1.4
0.3   1.0
0.4   1.2

+1.36695342e- 3   +9.71461661e- 4   +1.49629941e- 4
+3.99999999e- 1   +1.20000000e+ 0
+2.35687614e- 4   +9.71461661e- 4   +1.49629941e- 4
+3.99999999e- 1   +1.20000000e+ 0
+2.35687614e- 4   +6.99960719e- 5   +1.49629941e- 4
+3.99999999e- 1   +1.20000000e+ 0
+1.33517691e- 4   +6.99960719e- 5   +1.49629941e- 4
+3.34375001e- 1   +1.09375001e+ 0
+1.33517691e- 4   +6.99960719e- 5   +2.48749125e- 5
+3.34375001e- 1   +1.09375001e+ 0
+2.49822576e- 5   +6.99960719e- 5   +2.48749125e- 5
+3.82812500e- 1   +1.15312502e+ 0
+2.49822576e- 5   +1.29307009e- 5   +2.48749125e- 5
+3.82812500e- 1   +1.15312502e+ 0
+4.13457383e- 6   +1.29307009e- 5   +2.48749125e- 5
+3.47753908e- 1   +1.11660159e+ 0
+4.13457383e- 6   +1.29307009e- 5   +1.50262530e- 6
+3.52368169e- 1   +1.13032229e+ 0
+4.13457383e- 6   +3.33905754e- 6   +1.50262530e- 6
+3.66436772e- 1   +1.13829349e+ 0
+3.02300921e- 6   +3.33905754e- 6   +1.50262530e- 6
+3.66436772e- 1   +1.13829349e+ 0
+3.02300921e- 6   +3.04106077e- 7   +1.50262530e- 6
+3.66436772e- 1   +1.13829349e+ 0
+9.25855645e- 7   +3.04106077e- 7   +1.50262530e- 6
+3.60309988e- 1   +1.13065723e+ 0
+9.25855645e- 7   +3.04106077e- 7   +2.52192287e- 7
+3.60309988e- 1   +1.13065723e+ 0
+1.58032726e- 7   +3.04106077e- 7   +2.52192287e- 7
+3.63070875e- 1   +1.13569225e+ 0
+1.58032726e- 7   +1.10547128e- 8   +2.52192287e- 7
+3.60395197e- 1   +1.13434978e+ 0
+1.58032726e- 7   +1.10547128e- 8   +6.65574156e- 8
+3.61021515e- 1   +1.13283913e+ 0
+4.43762551e- 8   +1.10547128e- 8   +6.65574156e- 8
+3.61021515e- 1   +1.13283913e+ 0
+4.43762551e- 8   +1.10547128e- 8   +1.97336423e- 8
+3.61021515e- 1   +1.13283913e+ 0
+9.72654235e- 9   +1.10547128e- 8   +1.97336423e- 8
+3.61021515e- 1   +1.13283913e+ 0
+9.72654235e- 9   +1.10547128e- 8   +1.67431603e- 9
+3.61377098e- 1   +1.13313873e+ 0
+9.72654235e- 9   +4.65927780e- 9   +1.67431603e- 9
+3.61315422e- 1   +1.13351589e+ 0
+1.11289118e- 9   +4.65927780e- 9   +1.67431603e- 9
+3.61315422e- 1   +1.13351589e+ 0
+1.11289118e- 9   +6.16092122e-10   +1.67431603e- 9
+3.61122213e- 1   +1.13346039e+ 0
+1.11289118e- 9   +6.16092122e-10   +9.41459979e-10
+3.61201353e- 1   +1.13328569e+ 0
+1.46801349e-10   +6.16092122e-10   +9.41459979e-10
+3.61201353e- 1   +1.13328569e+ 0
```



```
+1.46801349e-10   +6.16092122e-10   +2.20032340e-10
+3.61113481e- 1   +1.13335916e+ 0
+1.46801349e-10   +9.38616407e-11   +2.20032340e-10
+3.61113481e- 1   +1.13335916e+ 0
+1.46801349e-10   +9.38616407e-11   +9.10996278e-11
+3.61149743e- 1   +1.13330172e+ 0
+1.72720310e-11   +9.38616407e-11   +9.10996278e-11
+3.61180443e- 1   +1.13338459e+ 0
+1.72720310e-11   +1.01777614e-11   +9.10996278e-11
+3.61139290e- 1   +1.13335116e+ 0
+1.72720310e-11   +1.01777614e-11   +1.38952322e-11
+3.61154806e- 1   +1.13333480e+ 0
+1.81694937e-12   +1.01777614e-11   +1.38952322e-11
+3.61154806e- 1   +1.13333480e+ 0
+1.81694937e-12   +1.01777614e-11   +3.37180284e-12
+3.61140937e- 1   +1.13333984e+ 0
+1.81694937e-12   +1.82664994e-12   +3.37180284e-12
+3.61140937e- 1   +1.13333984e+ 0
+1.81694937e-12   +1.82664994e-12   +1.47909462e-12
+3.61140937e- 1   +1.13333984e+ 0
+1.81694937e-12   +7.64943664e-14   +1.47909462e-12
+3.61149240e- 1   +1.13334489e+ 0
+4.69901895e-13   +7.64943664e-14   +1.47909462e-12
+3.61146256e- 1   +1.13333859e+ 0
+4.69901895e-13   +7.64943664e-14   +2.17881269e-13
+3.61146256e- 1   +1.13333859e+ 0
+1.02695630e-13   +7.64943664e-14   +2.17881269e-13
+3.61146256e- 1   +1.13333859e+ 0
+1.02695630e-13   +7.64943664e-14   +5.08620923e-14
+3.61146256e- 1   +1.13333859e+ 0
+1.80411242e-14   +7.64943664e-14   +5.08620923e-14
+3.61144375e- 1   +1.13333803e+ 0
+1.80411242e-14   +1.06720188e-14   +5.08620923e-14
+3.61144945e- 1   +1.13333882e+ 0
+1.80411242e-14   +1.06720188e-14   +1.74999904e-14
+3.61145459e- 1   +1.13333851e+ 0
+1.23512311e-15   +1.06720188e-14   +1.74999904e-14
+3.61145459e- 1   +1.13333851e+ 0
+1.23512311e-15   +1.06720188e-14   +4.99600361e-15
+3.61145347e- 1   +1.13333887e+ 0
+1.23512311e-15   +3.06699111e-15   +4.99600361e-15
+3.61145347e- 1   +1.13333887e+ 0
+1.23512311e-15   +3.06699111e-15   +1.11022302e-15
+3.61145347e- 1   +1.13333887e+ 0
+1.23512311e-15   +2.22044605e-16   +1.11022302e-15
+3.61145277e- 1   +1.13333872e+ 0
+2.22044605e-16   +2.22044605e-16   +1.11022302e-15
```

```
+3.61145407e- 1   +1.13333876e+ 0
+2.22044605e-16   +2.22044605e-16   +1.11022302e-16
+3.61145347e- 1   +1.13333881e+ 0
+2.77555756e-16   +1.11022302e-16   +1.11022302e-16
+3.61145478e- 1   +1.13333887e+ 0
+5.55111512e-17   +1.11022302e-16   +1.11022302e-16
+3.61145444e- 1   +1.13333882e+ 0
+5.55111512e-17   +6.93889390e-17   +1.38777878e-17
+3.61145440e- 1   +1.13333885e+ 0
+5.55111512e-17   +5.55111512e-17   +1.38777878e-17
+3.61145459e- 1   +1.13333887e+ 0
+6.93889390e-17   +5.55111512e-17   +1.38777878e-17
+3.61145459e- 1   +1.13333887e+ 0
+5.55111512e-17   +5.55111512e-17   +1.38777878e-17
+3.61145459e- 1   +1.13333887e+ 0
+5.55111512e-17   +5.55111512e-17   +1.38777878e-17
+3.61145459e- 1   +1.13333887e+ 0
min f

+1.38777878e-17
min point

+3.61145459e- 1   +1.13333887e+ 0
```


(v) The quasi-Newton example is an implementation of Huang's unified approach[10]. This is not the most efficient implementation but illust·ates all the quasi-Newton methods in one code. Moreover both Powell and Davidon line searches are implemented and can be interchanged by the choice of the BOOL . The constants rho. c1, c2, k1 and k2 are precisely those in Huang's original work. Rosenbrock's "banana region" has been used as the example. A run with Powell and a second one with Davidson as the line search are given.

```
PROGRAM huang
BEGIN
   [1:2] REAL x01,xm1;
   [1:4] REAL x02,xm2;
   REAL fm;
   [1:4] REAL d0,xo2;

   PROC f1=(REF [ ] REAL x)REAL:
   BEGIN
     REAL w:=(x[2]-x[1]*x[1]);
     100.0*w*w+(1.0-x[1])*(1.0-x[1])
   END;

   PROC g1=( INT i, REF [ ] REAL x) REAL:
   BEGIN
      CASE i IN
         -2.0*(200.0*x[1]*(x[2]-x[1]*x[1])+(1.0-x[1])),
         200.0*(x[2]-x[1]*x[1])
      ESAC
   END;

   PROC powell=(PROC(REF[ ]REAL)REAL f,
                REF [ ] REAL d,xk,REAL h,er)REAL:
   BEGIN
     REAL xa,xb,xc,fa,fb,fc,dd,ch,w,max,w1,w2,w3,min;
     INT n:=UPB xk,ul,i;
     [1:n] REAL xar;
     ul:=50;
     BOOL powout:=TRUE;
     FORMAT fq1=$l"X"3x3(+d.7de+zd3x)l,"F"3(3x+d.7de+zd)$;
      xa:=0.0;

      PROC tr=(REAL ex)VOID:
      BEGIN
        FOR i1 TO n DO xar[i1]:=xk[i1]+ex*d[i1] OD
      END;

      tr(xa);
      fa:=f(xar);
      xb:=xa+h;tr(xb);fb:=f(xar);
      IF fb<fa THEN xc:=xb+h;tr(xc);fc:=f(xar)
      ELSE
         xc:=xa-h;tr(xc);fc:=f(xar);
         w:=xc;xc:=xb;xb:=xa;xa:=w;
         w:=fc;fc:=fb;fb:=fa;fa:=w
      FI;
      TO ul WHILE
         (ABS(fa-fb)>er*(ABS fa +1.0e-20)) OR
         (ABS(fb-fc)>er*(ABS fb+1.0e-20))
      DO
         (powout!printf((fq1,xa,xb,xc,fa,fb,fc)));
         w1:=xb-xc;w2:=xc-xa;w3:=xa-xb;
         w:=(w1*fa+w2*fb+w3*fc);
         IF ABS w<1.0e-35 THEN print((newline,"FAIL2",newline))
```



```
         ELSE
            dd:=((xb+xc)*w1*fa+(xc+xa)*w2*fb+(xa+xb)*w3*fc)/w/2.0
         FI;
         ch:=(w1*fa+w2*fb+w3*fc)/w1/w2/w3;
         IF ch>0.0 THEN
            IF fc<fb THEN
               xa:=xb;fa:=fb;xb:=xc;fb:=fc;
               xc:=xb+h;tr(xc);fc:=f(xar)
            ELSE
               xc:=xb;fc:=fb;xb:=xa;fb:=fa;
               xa:=xb-h;tr(xa);fa:=f(xar)
            FI
         ELSE
            i:=1;max:=ABS(xa-dd);
            IF max< ABS(xb-dd) THEN max:=ABS(xb-dd);i:=2 FI ;
            IF max< ABS(xc-dd) THEN i:=3 FI ;
            CASE i IN
               (xa:=dd;tr(xa);fa:=f(xar)),
               (xb:=dd;tr(xb);fb:=f(xar)),
               (xc:=dd;tr(xc);fc:=f(xar))
            ESAC;
            IF xa>xb THEN
               w:=xa;xa:=xb;xb:=w;
               w:=fa;fa:=fb;fb:=w
            FI;
            IF xb>xc THEN
              w:=xb;xb:=xc;xc:=w;
              w:=fb;fb:=fc;fc:=w
            FI;
            IF xa>xb THEN
              w:=xa;xa:=xb;xb:=w;
              w:=fa;fa:=fb;fb:=w
            FI
         FI
      OD;
      i:=1;min:=fa;
       IF min>fb THEN min:=fb;i:=2 FI;
       IF min>fc THEN i:=3 FI;
       CASE i IN
          xa,xb,xc
       ESAC
   END;

   PROC davidon=(PROC(REF[ ]REAL)REAL f,
                 PROC(INT,REF[ ]REAL)REAL g,
                 REF[ ]REAL dd,xk,REAL hh,er)REAL:
   BEGIN
     REAL al,a,b,all,f0,fa,g0,ga,w1,w,z,lm,ac,fc;
     INT n:=UPB xk,ul1,ul2,ul1f,ul2f;
     ul1:=8;ul2:=20;
     BOOL b1:=TRUE,g0p;
      [1:n]REAL xar;
      FORMAT fd1=$l"A="+d.7de+zd7x,"B="+d.7de+zdl,
                  "FA="+d.7de+zd7x,"FB="+d.7de+zd$,
             fd2=$l"AL="+d.7de+zd11x,"FA=",+d.7de+zd11x,"GA="+d.7de+zd$;
      f0:=f(xk);
      g0:=0.0;
      FOR i1 TO n DO g0 PLUSAB g(i1,xk)*dd[i1] OD;
      g0p:= g0>=0.0;
      w:=0.0;
```

```
   FOR i1 TO n DO w PLUSAB dd[i1]*dd[i1] OD;
   al:=hh/sqrt(w);
   (g0p!al:=-al);
   FOR i1 TO ul1 WHILE b1 DO
      ul1f:=i1;
      FOR i1 TO n DO xar[i1]:=xk[i1]+al*dd[i1] OD;
      fa:=f(xar);
      ga:=0.0;
      FOR i1 TO n DO ga PLUSAB g(i1,xar)*dd[i1] OD;
      printf((fd2,al,fa,ga));
      IF (g0p!ga<0.0!ga>0.0) OR fa>f0 THEN b1:=FALSE
      ELSE al:=2*al
      FI
   OD;
   IF ul1f=ul1 THEN print((newline,"ul1 limit",newline)) FI;
   all:=al;
   a:=0.0;b:=al;
   FOR i1 TO  ul2 WHILE ABS(a-b)>er*(ABS(a+b)/2.0+0.5) DO
      ul2f:=i1;
      printf((fd1,a,b,f0,fa));
      z:=3.0*(f0-fa)/all+g0+ga;
      w:=sqrt(z*z-g0*ga);
      (g0p!w:=-w);
      lm:=all*(1.0-(ga+w-z)/(ga-g0+2*w))+a;
      IF
         FOR i1 TO n DO xar[i1]:=xk[i1]+lm*dd[i1] OD;
         gc:=0.0;
         FOR i1 TO n DO gc PLUSAB g(i1,xar)*dd[i1] OD;
         fc:=f(xar);
         (g0p AND gc<=0.0) OR (NOT g0p AND gc>=0.0)
      THEN
         b:=lm;ga:=gc;fa:=fc;all:=b-a
      ELSE
         a:=lm;g0:=gc;f0:=fc;all:=b-a
      FI
   OD;
   IF ul2f=ul2 THEN print((newline,"ul2 limit",newline)) FI;
   (a+b)/2.0
END;

PROC huang=(PROC(REF[ ]REAL)REAL f,
            PROC(INT,REF[ ]REAL)REAL g,
            REF[ ]REAL x0,xm,
            REAL er,hh,REF REAL fm)VOID:
BEGIN
   INT n:=UPB x0,ul;
   [1:n] REAL vec,p,gg,delg,delx,x1,x2,ved,gh,v1;
   [1:n,1:n] REAL mat,h,ht,mau;
   REAL rho,c1,c2,k1,k2,f1,f2,al,s,den1,den2;
   BOOL start:=TRUE,powl;
   powl:=FALSE;
   FORMAT fh1=$l10(+d.2de+zd2x)$,
          fh2=$l"RHO="+d.2de+zd,l"C1="+d.2de+zd5x,"C2="+d.2de[illegible],
              "K1="+d.2de+zd5x,"K2="+d.2de+zd$;
   FOR i1 TO n DO
      FOR i2 TO n DO
         (i1=i2!h[i1,i2]:=1.0!h[i1,i2]:=0.0)
      OD
   OD;
   f2:=f(x0);x2:=x0;
   f1:=f2+1.0;
   FOR i1 TO n DO gg[i1]:=g(i1,x2)OD;
   rho:=1.0;c1:=1.0;c2:=0.0;
   k1:=1.0;k2:=0.0;
   ul:=20;
   printf((fh2,rho,c1,c2,k1,k2));
   TO ul WHILE ABS(f1-f2)>er*(ABS f1+1.0e-20) DO
      f1:=f2;x1:=x2;
      IF start THEN
         FOR i1 TO n DO
            FOR i2 TO nDO
               ht[i1,i2]:=h[i1,i2]
            OD
         OD;
         start:=FALSE
      ELSE
         FOR i1 TO n DO gh[i1]:=g(i1,x1)OD;
         FOR i1 TO n DO delg[i1]:=gh[i1]-gg[i1] OD;
         gg:=gh;
         FOR i1 TO n DO
            s:=0.0;
            FOR i2 TO n DO
               s PLUSAB ht[i1,i2]*delg[i2]
            OD;
            v1[i1]:=s
         OD;
         FOR i1 TO n DO vec[i1]:=c1*delx[i1]+c2*v1[i1] OD;
         FOR i1 TO n DO ved[i1]:=k1*delx[i1]+k2*v1[i1] OD;
         s:=0.0;
         FOR i1 TO n DO s PLUSAB vec[i1]*delg[i1] OD;
         den1:=s;
         s:=0.0;
         FOR i1 TO n DO s PLUSAB ved[i1]*delg[i1] OD;
         den2:=s;
         FOR i1 TO n DO
            s:=0.0;
            FOR i2 TO n DO s PLUSAB h[i1,i2]*delg[i2] OD;
            v1[i1]:=s
         OD;
         FOR i1 TO n DO
            FOR i2 TO n DO
               mat[i1,i2]:=delx[i1]*vec[i2]
            OD
         OD;
         FOR i1 TO n DO
            FOR i2 TO n DO
               mau[i1,i2]:=v1[i1]*ved[i2]
            OD
         OD;
         FOR i1 TO n DO
            FOR i2 TO n DO
               h[i1,i2] PLUSAB rho*mat[i1,i2]/den1-mau[i1,i2]/den2;
               ht[i2,i1]:=h[i1,i2]
            OD
         OD
      FI;
      print((newline,"X"));
      printf((fh1,x1));
      print((newline,"G"));
      printf((fh1,gg));
```

```
        print((newline,"H"));
        FOR i1 TO n DO
           printf((fh1,h[i1, ]))
        OD;
        FOR i1 TO n DO
           s:=0.0;
           FOR i2 TO n DO s PLUSAB ht[i1,i2]*gg[i2] OD;
           p[i1]:=s
        OD;
        print((newline,"P"));
        printf((fh1,p));
        IF powl THEN
           al:=powell(f,p,x1,hh,er)
        ELSE
           al:=davidon(f,g,p,x1,hh,er)
        FI;
        FOR i1 TO n DO delx[i1]:=al*p[i1] OD;
        FOR i1 TO n DO x2[i1]:=x1[i1]+delx[i1] OD;
        f2:=f(x2)
     OD;
     xm:=x2;
    fm:=f2
  END;

  x01:=(-1.2,1.0);
  huang(f1,g1,x01,xm1,1.0e-04,0.1,fm);
  print((newline,"XM1=",xm1,space,space,"FM=",fm))
END
FINISH
```


Huang with Powell quadratic line search.

```
XM0=+1.0[illegible]e+ [illegible]
C1=+1.[illegible]e+ [illegible]      C2=+0.[illegible]e+ [illegible]
K1=+1.[illegible]e+ [illegible]      K2=+0.[illegible]e+ [illegible]
X
-1.2[illegible]e+ [illegible]   +1.[illegible]e+ [illegible]
G
-2.1[illegible]e+ 2   -[illegible].[illegible]e+ 1
H
+1.[illegible]e+ [illegible]   +[illegible]e+ [illegible]
+[illegible].[illegible]e+ [illegible]   +1.[illegible]e+ [illegible]
P
-2.1[illegible]e+ 2   -[illegible]e+ 1
A    -1.[illegible]e- 1     +[illegible]e+ [illegible]     -1.[illegible]e- 1
F    +1.[illegible]e+ 2     +[illegible]e+ 1               +2.[illegible]e+ 2
X    -1.[illegible]e- 1     -1.[illegible]e- 2             +[illegible]e+ [illegible]
F    +1.[illegible]e+ 2     +[illegible]e+ [illegible]     +[illegible]e+ 1
A    -1.[illegible]e- [illegible]   -[illegible]e- [illegible]   +[illegible]e+ [illegible]
F    +2.[illegible]e+ [illegible]   +[illegible]e+ [illegible]   +2.[illegible]e+ 1
A    -[illegible]e- 2     +[illegible]e+ [illegible]     +1.[illegible]e- 1
F    +2.[illegible]e+ [illegible]   +[illegible]e+ 1     +2.[illegible]e+ 2
A    -[illegible]e- 2     -[illegible]e- [illegible]     -[illegible]e- [illegible]
F    +2.[illegible]e+ 2   +1.[illegible]e+ [illegible]   +2.[illegible]e+ 1
X    -[illegible]e- 2     +[illegible]e+ [illegible]     -[illegible]e- 2
F    +1.[illegible]e+ [illegible]   +[illegible]e+ 1     +1.[illegible]e+ 2
A    -[illegible]e- 2     -1.[illegible]e- [illegible]   -[illegible]e+ [illegible]
F    +1.[illegible]e+ [illegible]   +1.[illegible]e+ 1   +[illegible]e+ [illegible]
X    -1.[illegible]e- [illegible]   -[illegible]e- [illegible]   +[illegible]e+ [illegible]
F    +1.[illegible]e+ 1   +[illegible]e+ [illegible]     +[illegible]e- 1
A    -1.[illegible]e- [illegible]   -[illegible]e- [illegible]   -[illegible]e- [illegible]
F    +1.[illegible]e+ 1   +[illegible]e+ [illegible]     +[illegible]e+ [illegible]
A    -[illegible]e- [illegible]     -[illegible]e- [illegible]   -7.[illegible]e- [illegible]
F    +[illegible]e+ [illegible]     +[illegible]e+ [illegible]   +[illegible]e+ [illegible]
X    -[illegible]e- [illegible]     -7.[illegible]e- [illegible] -7.[illegible]e- [illegible]
F    +[illegible]e+ [illegible]     +[illegible]e+ [illegible]   +[illegible]e+ [illegible]
X    -7.[illegible]e- [illegible]   -7.[illegible]e- [illegible] -7.[illegible]e- [illegible]
F    +[illegible]e+ [illegible]     +[illegible]e+ [illegible]   +[illegible]e+ [illegible]
X
-1.[illegible]e+ [illegible]   +1.[illegible]e+ [illegible]   1)
G
-[illegible]e- 1   +1.[illegible]e+ [illegible]
H
+1.[illegible]e- 1   -[illegible]e- 1
```

[illegible]

F

[illegible]

A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]

X

[illegible] 2)

S

[illegible]

n

[illegible]

[illegible]

P

[illegible]

A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]

X

[illegible] 3)

G

[illegible]

n

[illegible]

[illegible]

P

[illegible]

X	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
X	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
X	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
X	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
X	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]

F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
X	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
X	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
X	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
X	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
X	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
X	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]
F	[illegible]	[illegible]	[illegible]
A	[illegible]	[illegible]	[illegible]

```
F    +2.11[illegible]e+ 0    +2.1[illegible]e+ 0    +2.1[illegible]e+ 0
X    -1.17[illegible]e+ 1    -[illegible]e+ 0    -3.71[illegible]e+ 0
F    +2.1713[illegible]e+ 1    +2.11[illegible]e+ 0    +2.1[illegible]e+ 0
X    -4.[illegible]e+ 0    -[illegible]e+ 0    -3.71[illegible]e+ 0
F    +2.[illegible]e+ 0    +2.11[illegible]e+ 0    +2.14[illegible]e+ 0
X    -2.[illegible]e+ 0    -4.[illegible]e+ 0    -3.[illegible]e+ 0
F    +9.1561412e+ 0    +2.0[illegible]e+ 0    +2.11[illegible]e+ 0
X    -4.3[illegible]e+ 0    -4.[illegible]e+ 0    -3.[illegible]e+ 0
F    +1.[illegible]e+ 0    +2.[illegible]e+ 0    +2.11[illegible]e+ 0
X    -7.[illegible]e+ 0    -4.[illegible]e+ 0    -[illegible]e+ 0
F    +2.9[illegible]174e+ 0    +1.9[illegible]e+ 0    +2.[illegible]e+ 0
X    -4.[illegible]e+ 0    -4.[illegible]e+ 0    -4.[illegible]e+ 0
F    +1.[illegible]e+ 0    +1.[illegible]e+ 0    +2.[illegible]e+ 0
X    -5.2[illegible]e+ 0    -4.[illegible]e+ 0    -4.[illegible]e+ 0
F    +1.[illegible]e+ 0    +1.[illegible]e+ 0    +1.[illegible]e+ 0
X    -5.[illegible]e+ 0    -5.[illegible]e+ 0    -4.[illegible]e+ 0
F    +1.[illegible]e+ 0    +1.[illegible]e+ 0    +1.[illegible]e+ 0
X    -5.[illegible]e+ 0    -5.[illegible]e+ 0    -4.[illegible]e+ 0
F    +1.[illegible]e+ 0    +1.[illegible]e+ 0    +1.[illegible]e+ 0
X    -5.3101[illegible]e+ 0    -5.[illegible]e+ 0    -5.21[illegible]e+ 0
F    +1.[illegible]1547e+ 0    +1.[illegible]e+ 0    +1.[illegible]e+ 0
X    -5.71[illegible]e+ 0    -5.[illegible]e+ 0    -5.[illegible]e+ 0
F    +1.[illegible]1547e+ 0    +1.[illegible]e+ 0    +1.[illegible]e+ 0
X
-[illegible]e- 1    +7.[illegible]e- 2    4)
G
-7.[illegible]e+ 0    -[illegible]e+ 0
H
+5.4[illegible]e- 2    -4.[illegible]e- 2
-2.17e- 2    +1.4[illegible]e- 1
P
+2.[illegible]e- 2    -2.[illegible]e- 1
X    -1.[illegible]e- 1    +0.[illegible]e+ 0    +1.[illegible]e- 1
F    +1.[illegible]e+ 0    +1.[illegible]e+ 0    +2.[illegible]e+ 0
X    -1.31[illegible]e- 1    -1.[illegible]e- 1    +0.[illegible]e+ 0
F    +1.[illegible]e+ 0    +1.[illegible]e+ 0    +1.[illegible]e+ 0
X    -1.725[illegible]e- 1    -1.31[illegible]e- 1    -1.[illegible]e- 1
F    +1.[illegible]e+ 0    +1.[illegible]e+ 0    +1.[illegible]e+ 0
X
-3.41e- 1    +1.1[illegible]e- 1
G
-3.[illegible]e+ 0    -1.19e+ 0
H
+4.[illegible]e- 2    -2.51e- 2
-4.41e- 2    +3.51e- 2
P
-[illegible].71e- 2    +5.[illegible]e- 2
X    -1.[illegible]e- 1    +0.[illegible]e+ 0    +1.[illegible]e- 1
F    +1.77[illegible]3795e+ 0    +1.[illegible]e+ 0    +1.[illegible]e+ 0
X    -6.7513[illegible]e+ 0    -1.[illegible]e- 1    +0.[illegible]e+ 0
F    +1.1[illegible]2122e+ 1    +1.77[illegible]e+ 0    +1.[illegible]e+ 0
X    -5.142[illegible]e- 1    -1.[illegible]e- 1    +0.[illegible]e+ 0
F    +1.[illegible]e+ 0    +1.7[illegible]795e+ 0    +1.[illegible]e+ 0
X    -7.12[illegible]e+ 0    -5.1[illegible]e- 1    -1.[illegible]e- 1
F    +1.[illegible]e+ 1    +1.[illegible]e+ 0    +1.77[illegible]e+ 0
X    -6.[illegible]e- 1    -5.1[illegible]e- 1    -1.[illegible]e- 1
F    +1.[illegible]e+ 0    +1.[illegible]e+ 0    +1.77[illegible]e+ 0
X    -6.[illegible]e+ 0    -[illegible]e- 1    -[illegible]e- 1
F    +9.[illegible]e+ 0    +1.[illegible]e+ 0    +1.[illegible]e+ 0
X    -1.[illegible]e+ 0    -[illegible]e- 1    -3.1[illegible]e- 1
```



```
F    +1.5831153e+ 0    +1.547[illegible]e+ 0    +1.6[illegible]e+ 0
X    -4.4022[illegible]e+ 0    -1.0035984e+ 0    -6.9135612e- 1
F    +3.3437364e+ 0    +1.5831153e+ 0    +1.647[illegible]e+ 0
X    -1.4317993e+ 0    -1.0035984e+ 0    -6.9135612e- 1
F    +1.5090088e+ 0    +1.5831153e+ 0    +1.647[illegible]e+ 0
X    -3.2773933e+ 0    -1.4317993e+ 0    -1.0035984e+ 0
F    +1.7201947e+ 0    +1.5090088e+ 0    +1.5831153e+ 0
X    -1.2995974e+ 0    -1.4317993e+ 0    -1.0035984e+ 0
F    +1.4586411e+ 0    +1.5090088e+ 0    +1.5831153e+ 0
X    -2.4185276e+ 0    -1.2995974e+ 0    -1.4317993e+ 0
F    +1.4044427e+ 0    +1.4586611e+ 0    +1.5090088e+ 0
X    -2.4186876e+ 0    -2.1129701e+ 0    -1.2995974e+ 0
F    +1.4044427e+ 0    +1.4513656e+ 0    +1.4586411e+ 0
X    -2.1216628e+ 0    -2.1129701e+ 0    -1.2995974e+ 0
F    +1.4513266e+ 0    +1.4513656e+ 0    +1.4586411e+ 0
X
-1.55e- 1    -1.02e- 2
G
-4.43e+ 0    -5.65e+ 0
H
+6.36e- 2    -4.34e- 2
-3.07e- 2    +2.54e- 2
P
-7.15e- 2    +1.17e- 2
X    -1.0000000e- 1    +0.0000000e+ 0    +1.0000000e- 1
F    +1.4231718e+ 0    +1.4513134e+ 0    +1.4754732e+ 0
X    -2.3213051e+ 0    -1.0000000e- 1    +0.0000000e+ 0
F    +1.1195026e+ 0    +1.4231718e+ 0    +1.4513134e+ 0
X    -2.2550598e+ 0    -2.3213251e+ 0    -1.0000000e- 1
F    +1.1215410e+ 0    +1.1195025e+ 0    +1.4231714e+ 0
X    -2.2550598e+ 0    -2.6358682e+ 0    -2.3213051e+ 0
F    +1.1215410e+ 0    +1.1122394e+ 0    +1.1195026e+ 0
X    -2.4164174e+ 0    -2.6058652e+ 0    -2.3213051e+ 0
F    +1.1124112e+ 0    +1.1129384e+ 0    +1.1195026e+ 0
X
+3.25e- 2    -4.11e- 2
G
-1.38e+ 0    -2.43e+ 0
H
+4.10e- 2    -3.97e- 2
+8.47e- 4    +2.12e- 2
P
-6.39e- 2    -1.24e- 1
X    -1.0000000e- 1    +0.0000000e+ 0    +1.0000000e- 1
F    +1.0142999e+ 0    +1.1129018e+ 0    +1.2409264e+ 0
X    -3.8511766e- 1    -1.0000000e- 1    +0.0000000e+ 0
F    +8.9952733e- 1    +1.0142999e+ 0    +1.1129018e+ 0
X    -3.9621856e- 1    -3.8511766e- 1    -1.0000000e- 1
F    +8.3922624e- 1    +8.9952733e- 1    +1.0142999e+ 0
X    -4.0045407e- 1    -3.9621856e- 1    -3.8511766e- 1
F    +8.8919951e- 1    +8.3922624e- 1    +8.9952733e- 1
X
+5.84e- 2    +8.58e- 3
G
-2.00e+ 0    +1.03e+ 0
H
+6.51e- 2    +7.00e- 3
-7.63e- 3    +4.74e- 3
P
-1.36e- 1    -9.11e- 3
```

```
X    -1.0000000e- 1   +0.0000000e+ 0   +1.0000000e- 1
F    +8.6246595e- 1   +8.6312946e- 1   +9.1610717e- 1
X    -2.8403495e+ 1   -1.0000000e- 1   +0.0000000e+ 0
F    +2.4496761e+ 4   +8.6248595e- 1   +8.5919946e- 1
X    -1.0000000e- 1   -5.4353440e- 2   +0.0000000e+ 0
F    +8.6246595e- 1   +8.7454902e- 1   +8.8919946e- 1
X    -1.4575359e+ 1   -1.0000000e- 1   -5.4353440e- 2
F    +1.7365537e+ 3   +8.6248595e- 1   +8.7464692e- 1
X    -1.0000000e- 1   -9.3310430e- 2   -5.4353440e- 2
F    +8.6246595e- 1   +8.6426900e- 1   +8.7464692e- 1
X    -1.0675310e+ 1   -1.0000000e- 1   -9.3302430e- 2
F    +5.0615218e+ 2   +8.6248595e- 1   +8.6426900e- 1
X    -1.2585966e- 1   -1.0000000e- 1   -9.3302430e- 2
F    +8.5561652e- 1   +8.6248595e- 1   +8.6425900e- 1
X    -8.7246352e+ 0   -1.2585966e- 1   -1.0000000e- 1
F    +2.2930908e+ 2   +8.5561652e- 1   +8.6246595e- 1
X    -1.5562239e- 1   -1.2585966e- 1   -1.0000000e- 1
F    +8.4774048e- 1   +8.5561652e- 1   +8.6246595e- 1
X    -7.3599369e+ 0   -1.5562239e- 1   -1.2585966e- 1
F    +1.1940570e+ 2   +8.4774048e- 1   +8.5561652e- 1
X    -1.9545548e- 1   -1.5562239e- 1   -1.2585966e- 1
F    +8.3547713e- 1   +8.4774048e- 1   +8.5561652e- 1
X    -5.2945984e+ 0   -1.9545548e- 1   -1.5562239e- 1
F    +4.9275025e+ 1   +8.3547713e- 1   +8.4774048e- 1
X    -2.6511844e- 1   -1.9545548e- 1   -1.5562239e- 1
F    +8.1968519e- 1   +8.3547713e- 1   +8.4774048e- 1
X    -4.4945100e+ 0   -2.6511844e- 1   -1.9545548e- 1
F    +1.7194122e+ 1   +8.1968519e- 1   +8.3547713e- 1
X    -3.6588668e- 1   -2.6511844e- 1   -1.9545548e- 1
F    +7.9375305e- 1   +8.1968519e- 1   +8.3547713e- 1
X    -3.2009231e+ 0   -3.6588668e- 1   -2.6511844e- 1
F    +4.9121359e+ 0   +7.9375305e- 1   +8.1968519e- 1
X    -5.3651107e- 1   -3.6588668e- 1   -2.6511844e- 1
F    +7.5394459e- 1   +7.9375305e- 1   +8.1968519e- 1
X    -2.1232708e+ 0   -5.3651107e- 1   -3.6588668e- 1
F    +1.3440396e+ 0   +7.5394459e- 1   +7.9375305e- 1
X    -7.8922068e- 1   -5.3651107e- 1   -3.6588668e- 1
F    +7.1791012e- 1   +7.5394459e- 1   +7.9375305e- 1
X    -1.4090567e+ 0   -7.8922068e- 1   -5.3651107e- 1
F    +7.4055153e- 1   +7.1791012e- 1   +7.5394458e- 1
X    -1.4090567e+ 0   -1.0913607e+ 0   -7.8922068e- 1
F    +7.4055153e- 1   +6.8932675e- 1   +7.1791012e- 1
X    -1.0230416e+ 0   -1.0913607e+ 0   -7.8922068e- 1
F    +6.8821112e- 1   +6.8932675e- 1   +7.0791012e- 1
X    -1.0556275e+ 0   -1.0230416e+ 0   -1.0913607e+ 0
F    +6.8659905e- 1   +6.8821112e- 1   +6.8932675e- 1
X    -1.0556275e+ 0   -1.0495228e+ 0   -1.0230416e+ 0
F    +6.8659906e- 1   +6.8595580e- 1   +6.8821112e- 1
X
+2.04e- 1   +1.61e- 2
G
+3.06e- 1   -4.46e+ 0
H
+8.30e- 2   +8.17e- 3
+2.04e- 2   +6.59e- 3
P
-6.97e- 2   -2.62e- 2
X    -1.0000000e- 1   +0.0000000e+ 0   +1.0000000e- 1
F    +6.7783058e- 1   +6.8858580e- 1   +6.9989104e- 1
X    -2.2053971e+ 0   -1.0000000e- 1   +0.0000000e+ 0
```



```
F    +6.1701629e- 1   +6.7783058e- 1   +6.8858580e- 1
X    -2.2053971e+ 0   -1.4758219e+ 0   -1.0000000e- 1
F    +6.1701629e- 1   +5.9723395e- 1   +6.7783058e- 1
X    -2.2053971e+ 0   -1.4758210e+ 0   -1.3695516e+ 0
F    +6.1701629e- 1   +5.9723395e- 1   +5.9767626e- 1
X    -1.4758210e+ 0   -1.4545324e+ 0   -1.3695516e+ 0
F    +5.9723395e- 1   +5.9721947e- 1   +5.9767626e- 1
X
+3.05e- 1   +5.93e- 2
G
+2.74e+ 0   -6.77e+ 0
H
+7.05e- 2   -1.32e- 2
+2.36e- 2   +7.97e- 3
P
-7.78e- 2   -9.01e- 2
X    -1.0000000e- 1   +0.0000000e+ 0   +1.0000000e- 1
F    +5.5982523e- 1   +5.9721815e- 1   +6.3917346e- 1
X    -9.4430393e- 1   -1.0000000e- 1   +0.0000000e+ 0
F    +3.9341220e- 1   +5.5982523e- 1   +5.9721815e- 1
X    -1.3512380e+ 0   -9.4430393e- 1   -1.0000000e- 1
F    +3.7535555e- 1   +3.9341220e- 1   +5.5982523e- 1
X    -1.2927790e+ 0   -1.0812380e+ 0   -9.4430393e- 1
F    +3.6509513e- 1   +3.7535555e- 1   +3.9341220e- 1
X    -1.4559703e+ 0   -1.2927790e+ 0   -1.0812380e+ 0
F    +3.6116766e- 1   +3.6509513e- 1   +3.7535555e- 1
X    -1.5518472e+ 0   -1.4559703e+ 0   -1.2927790e+ 0
F    +3.6034724e- 1   +3.6116766e- 1   +3.6509513e- 1
X    -1.6019046e+ 0   -1.5518472e+ 0   -1.4559703e+ 0
F    +3.6015593e- 1   +3.6034724e- 1   +3.6116766e- 1
X    -1.6222056e+ 0   -1.6019046e+ 0   -1.5518472e+ 0
F    +3.6013216e- 1   +3.6015593e- 1   +3.6034724e- 1
X
+4.32e- 1   +2.06e- 1
G
-4.46e+ 0   +3.27e+ 0
H
+1.26e- 1   +9.73e- 2
+6.27e- 2   +5.11e- 2
P
-2.94e- 1   -2.00e- 1
X    -1.0000000e- 1   +0.0000000e+ 0   +1.0000000e- 1
F    +3.3718176e- 1   +3.6013077e- 1   +4.1417952e- 1
X    -4.9533741e+ 0   -1.0000000e- 1   +0.0000000e+ 0
F    +5.2424525e+ 2   +3.3718176e- 1   +3.6013077e- 1
X    -1.0000000e- 1   -6.1649675e- 2   +0.0000000e+ 0
F    +3.3718176e- 1   +3.2698131e- 1   +3.6013077e- 1
X    -1.2655427e+ 0   -1.0000000e- 1   -6.1649675e- 2
F    +3.7728831e+ 0   +3.3718176e- 1   +3.2698131e- 1
X    -1.7269202e- 1   -1.0000000e- 1   -6.1649675e- 2
F    +2.7303863e- 1   +3.3718176e- 1   +3.2698131e- 1
X    -6.9513598e- 1   -1.7269202e- 1   -1.0000000e- 1
F    +4.9625364e- 1   +2.7303863e- 1   +3.3718176e- 1
X    -2.9257455e- 1   -1.7269202e- 1   -1.0000000e- 1
F    +2.3391628e- 1   +2.7303863e- 1   +3.3718176e- 1
X    -4.5182278e- 1   -2.9257455e- 1   -1.7269202e- 1
F    +2.4159280e- 1   +2.3391628e- 1   +2.7303863e- 1
X    -4.5182278e- 1   -3.5423623e- 1   -2.9257455e- 1
F    +2.4159280e- 1   +2.2675233e- 1   +2.3391628e- 1
X    -3.5789051e- 1   -3.5423623e- 1   -2.9257455e- 1
```

```
F    +2.2560441e- 1    +2.2475273e- 1    +2.3321628e- 1
X    -3.6223302e- 1    -3.5732051e- 1    -3.5425426e- 1
F    +2.2660041e- 1    +2.2603441e- 1    +2.2675233e- 1
X    -3.6228302e- 1    -3.6177547e- 1    -3.5729251e- 1
F    +2.2660041e- 1    +2.2655920e- 1    +2.2666441e- 1
X
+5.35e- 1   +2.78e- 1
G
+1.58e+ 0   -2.32e+ 0
H
+9.72e- 2   +7.79e- 2
+8.50e- 2   +7.15e- 2
P
-4.42e- 2   -4.32e- 2
X    -1.0000000e- 1    +0.0000000e+ 0    +1.0000000e- 1
F    +2.2347624e- 1    +2.2655974e- 1    +2.2981235e- 1
X    -1.3269513e+ 0    -1.0000000e- 1    +0.0000000e+ 0
F    +2.1322840e- 1    +2.2357494e- 1    +2.2655974e- 1
X    -1.8289513e+ 0    -1.1965443e+ 0    -1.0000000e- 1
F    +2.1322840e- 1    +2.0592099e- 1    +2.2347424e- 1
X    -1.4269513e+ 0    -1.1965443e+ 0    -1.1477250e+ 0
F    +2.1322840e- 1    +2.0532089e- 1    +2.0556759e- 1
X    -1.2115945e+ 0    -1.1265443e+ 0    -1.1477850e+ 0
F    +2.0561415e- 1    +2.0552089e- 1    +2.0556703e- 1
X
+5.92e- 1   +3.51e- 1
G
+3.87e+ 0   -3.96e+ 0
H
+3.69e- 2   +1.89e- 2
+4.82e- 2   +5.55e- 2
P
-4.79e- 2   -6.72e- 2
X    -1.0000000e- 1    +0.0000000e+ 0    +1.0000000e- 1
F    +1.9777525e- 1    +2.0561372e- 1    +2.1394250e- 1
X    -1.3023669e+ 0    -1.0000000e- 1    +0.0000000e+ 0
F    +1.1042800e- 1    +1.9777525e- 1    +2.0561372e- 1
X    -2.4583653e+ 0    -1.8023669e+ 0    -1.0000000e- 1
F    +8.5401644e- 2    +1.1042090e- 1    +1.9777523e- 1
X    -3.9226654e+ 0    -2.6593633e+ 0    -1.8022692e+ 0
F    +5.8225955e- 2    +8.5401646e- 2    +1.1042800e- 1
X    -4.2130209e+ 0    -3.9226654e+ 0    -2.6523633e+ 0
F    +6.9007853e- 2    +5.8225955e- 2    +8.5401644e- 2
X    -4.2130209e+ 0    -3.9226654e+ 0    -3.3063171e+ 0
F    +6.9007853e- 2    +6.3225955e- 2    +6.8257416e- 2
X    -3.9220654e+ 0    -3.9127942e+ 0    -3.8063171e+ 0
F    +6.8225955e- 2    +6.8224950e- 2    +6.8257616e- 2
X
+7.80e- 1   +5.94e- 1
G
+3.94e+ 0   -2.51e+ 0
H
+1.34e- 1   +1.55e- 1
+1.79e- 1   +2.18e- 1
P
+2.56e- 2   -1.48e- 3
X    -1.0000000e- 1    +0.0000000e+ 0    +1.0000000e- 1
F    +5.9141467e- 2    +6.8224947e- 2    +8.1179213e- 2
X    -2.9836055e- 1    -1.0000000e- 1    +0.0000000e+ 0
F    +5.2240767e- 2    +5.9141487e- 2    +6.8224947e- 2
```



```
X    -2.9142597e- 1    -2.9836055e- 1    -1.0000000e- 1
F    +5.2249459e- 2    +5.2240767e- 2    +5.9141467e- 2
X
+7.72e- 1   +5.94e- 1
G
-1.55e- 2   -2.55e- 1
H
+1.02e- 1   +1.57e- 1
+1.41e- 1   +2.21e- 1
P
-4.17e- 2   -6.54e- 2
X    -1.0000000e- 1    +0.0000000e+ 0    +1.0000000e- 1
F    +5.0537931e- 2    +5.2244768e- 2    +5.4202821e- 2
X    -4.1309932e+ 0    -1.0000000e- 1    +0.0000000e+ 0
F    +7.4267325e- 2    +5.0357971e- 2    +5.2246762e- 2
X    -1.5946946e+ 0    -1.0000000e- 1    +0.0000000e+ 0
F    +2.7939913e- 2    +5.0357931e- 2    +5.2246764e- 2
X    -3.7572543e+ 0    -1.5948605e+ 0    -1.0000000e- 1
F    +5.4784517e- 2    +2.7939915e- 2    +5.0537931e- 2
X    -1.9477558e+ 0    -1.5948605e+ 0    -1.0000000e- 1
F    +2.5684153e- 2    +2.7039913e- 2    +5.0537931e- 2
X    -2.2579052e+ 0    -1.9477654e+ 0    -1.5946946e+ 0
F    +2.7032472e- 2    +2.5846155e- 2    +2.7439913e- 2
X    -2.2579062e+ 0    -2.2243582e+ 0    -1.9477653e+ 0
F    +2.7030472e- 2    +2.4458507e- 2    +2.5584153e- 2
X    -2.2553764e+ 0    -2.2243582e+ 0    -1.9477652e+ 0
F    +2.4451944e- 2    +2.4453607e- 2    +2.5584153e- 2
X    -2.2563506e+ 0    -2.2359784e+ 0    -2.2243552e+ 0
F    +2.4445512e- 2    +2.4451944e- 2    +2.4458092e- 2
X    -2.2663646e+ 0    -2.2424273e+ 0    -2.2359724e+ 0
F    +2.4445512e- 2    +2.4445374e- 2    +2.4451944e- 2
X
+9.45e- 1   +7.42e- 1
G
+2.54e+ 0   -1.62e+ 0
H
+1.25e- 1   +3.02e- 1
+3.20e- 1   +5.02e- 1
P
-2.26e- 2   -4.52e- 2
X    -1.0000000e- 1    +0.0000000e+ 0    +1.0000000e- 1
F    +2.3152265e- 2    +2.4445362e- 2    +2.5905721e- 2
X    -2.0116269e+ 0    -1.0000000e- 1    +0.0000000e+ 0
F    +7.7425948e- 3    +2.3152685e- 2    +2.4445362e- 2
X    -2.7227679e+ 0    -2.0118249e+ 0    -1.0000000e- 1
F    +4.8599623e- 3    +7.7425948e- 3    +2.3152855e- 2
X    -3.4947801e+ 0    -2.7227679e+ 0    -2.0118269e+ 0
F    +2.1570103e- 3    +4.8599623e- 3    +7.7425948e- 3
X    -5.0227013e+ 0    -3.4947801e+ 0    -2.7227679e+ 0
F    +2.7205148e- 4    +2.1575103e- 3    +4.8599623e- 3
X    -5.5615932e+ 0    -5.0227013e+ 0    -3.4947901e+ 0
F    +2.4355395e- 4    +2.7207146e- 4    +2.1570103e- 3
X    -5.5615932e+ 0    -5.3370684e+ 0    -5.0227013e+ 0
F    +2.4355395e- 4    +1.8325190e- 4    +2.7205148e- 4
X    -5.5615932e+ 0    -5.3370684e+ 0    -5.3290623e+ 0
F    +2.4355395e- 4    +1.8325190e- 4    +1.9330295e- 4
X    -5.3370564e+ 0    -5.3352515e+ 0    -5.3290623e+ 0
F    +1.8325150e- 4    +1.8324675e- 4    +1.8330295e- 4
X
+9.32e- 1   +9.63e- 1
```

```
G
+4.25e- 1  -2.21e- 1
H
+4.70e- 1  +6.96e- 1
+7.05e- 1  +1.24e+ 0
P
+1.35e- 2  +2.04e- 2
X    -1.0000000e- 1    +0.0000000e+ 0    +1.0000000e- 1
F    +1.0594475e- 4    +1.8724876e- 4    +3.4609825e- 4
X    -1.4036598e- 1    -1.0000000e- 1    +0.0000000e+ 0
F    +9.4659764e- 5    +1.0594475e- 4    +1.8324876e- 4
X    -1.4155209e- 1    -1.4036598e- 1    -1.0000000e- 1
F    +9.4652104e- 5    +9.4659764e- 5    +1.0594475e- 4
X
+9.90e- 1  +9.30e- 1
G
+6.32e- 2  -4.17e- 2
H
+2.04e- 1  +3.99e- 1
+5.22e- 1  +7.50e- 1
P
-3.05e- 3  -5.09e- 3
X    -1.0000000e- 1    +0.0000000e+ 0    +1.0000000e- 1
F    +9.2531713e- 5    +9.4652104e- 5    +1.0427692e- 4
X    -3.0439554e+ 0    -1.0000000e- 1    +0.0000000e+ 0
F    +4.9574733e- 7    +9.2531713e- 5    +9.4652104e- 5
X    -3.1614468e+ 0    -3.0439554e+ 0    -1.0000000e- 1
F    +2.4482125e- 7    +4.9574733e- 7    +9.2531713e- 5
X    -3.2106155e+ 0    -3.1614468e+ 0    -3.0439554e+ 0
F    +2.2010005e- 7    +2.4482125e- 7    +4.9574733e- 7
X    -3.2129515e+ 0    -3.2106185e+ 0    -3.1514468e+ 0
F    +2.2005249e- 7    +2.2010005e- 7    +2.4482125e- 7
X    -3.2129515e+ 0    -3.2121742e+ 0    -3.2106155e+ 0
F    +2.2005249e- 7    +2.2021701e- 7    +2.2010005e- 7
X    -3.2121742e+ 0    -3.2106185e+ 0    -3.1106155e+ 0
F    +2.2021701e- 7    +2.2010005e- 7    +3.1720041e- 7
X    -3.2121742e+ 0    -3.2106185e+ 0    -3.2077693e+ 0
F    +2.2021701e- 7    +2.2010005e- 7    +2.2025502e- 7
X    -3.2121742e+ 0    -3.2106185e+ 0    -3.2100461e+ 0
F    +2.2021701e- 7    +2.2010005e- 7    +2.2022660e- 7
X    -3.2121742e+ 0    -3.2111455e+ 0    -3.2106155e+ 0
F    +2.2021701e- 7    +2.2004205e- 7    +2.2010005e- 7
X    -3.2112054e+ 0    -3.2111455e+ 0    -3.2106155e+ 0
F    +2.1991197e- 7    +2.2004206e- 7    +2.2010005e- 7
X    -3.3112054e+ 0    -3.2112054e+ 0    -3.2111455e+ 0
F    +3.0961351e- 7    +2.1991197e- 7    +2.2004206e- 7
X    -3.2465402e+ 0    -3.2112054e+ 0    -3.2111455e+ 0
F    +2.3044151e- 7    +2.1991197e- 7    +2.2004206e- 7
X    -3.2267352e+ 0    -3.2112054e+ 0    -3.2111455e+ 0
F    +2.2185177e- 7    +2.1991197e- 7    +2.2004206e- 7
X    -3.2185547e+ 0    -3.2112054e+ 0    -3.2111455e+ 0
F    +2.2033247e- 7    +2.1991197e- 7    +2.2004206e- 7
X    -3.2148083e+ 0    -3.2112054e+ 0    -3.2111455e+ 0
F    +2.2019439e- 7    +2.1991197e- 7    +2.2004206e- 7
X    -3.2129484e+ 0    -3.2112054e+ 0    -3.2111455e+ 0
F    +2.1994208e- 7    +2.1991197e- 7    +2.2004206e- 7
X
+1.00e+ 0  +1.00e+ 0
G
+1.82e- 2  -9.36e- 2
```



```
H
+4.92e- 1  +9.76e- 1
+9.78e- 1  +1.95e+ 0
P
+7.23e- 5  +7.59e- 5
X    -1.0000000e- 1    +0.0000000e+ 0    +1.0000000e- 1
F    +1.7636975e- 7    +2.1991197e- 7    +2.6314574e- 7
X    -9.4403523e- 1    -1.0000000e- 1    +0.0000000e+ 0
F    +1.8644677e- 9    +1.7633975e- 7    +2.1991197e- 7
X    -9.6403523e- 1    -9.4946217e- 1    -1.0000000e- 1
F    +1.8644677e- 9    +1.0959570e- 9    +1.7636975e- 7
X    -9.4403623e- 1    -9.4946217e- 1    -9.4307402e- 1
F    +1.8644677e- 9    +1.0057400e- 9    +1.9307822e- 7
XM1= +9.9995709e -1  +9.0091530e -1    FM= +1.8029375e -9
```

Huang-Davidon

```
huang

RHO=+1.00e+ 0
C1=+1.00e+ 0      C2=+0.00e+ 0
K1=+1.00e+ 0      K2=+0.00e+ 0
X
-1.20e+ 0  +1.00e+ 0
G
-2.16e+ 2  -8.80e+ 1
H
+1.00e+ 0  +0.00e+ 0
+0.00e+ 0  +1.00e+ 0
P
-2.16e+ 2  -8.80e+ 1
AL=-4.2942839e- 4              FA=+7.9973960e+ 0              GA=+2.2237[illegible]0e+ 4
AL=-8.5885678e- 4              FA=+4.2683773e+ 0              GA=-3.9349[illegible]4e+ 3
A=+0.0000000e+ 0          B=-8.5885678e- 4
FA=+2.4200001e+ 1          FB=+4.2683773e+ 0
A=-7.8769570e- 4          B=-8.5885678e- 4
FA=+4.1281000e+ 0          FB=+4.2683773e+ 0
X
-1.03e+ 0  +1.07e+ 0
G
-7.05e- 1  +1.63e+ 0
H
+1.46e- 1  -3.49e- 1
-3.56e- 1  +8.55e- 1
P
-6.83e- 1  +1.64e+ 0
AL=-5.6361341e- 2              FA=+3.9706415e+ 0              GA=+2.36299[illegible]9e+ 0
AL=-1.1272268e- 1              FA=+3.8711871e+ 0              GA=+1.07453[illegible]0e+ 0
AL=-2.2544537e- 1              FA=+3.9755196e+ 0              GA=-3.3857[illegible]0e+ 0
A=+0.0000000e+ 0          B=-2.2544537e- 1
FA=+4.1280980e+ 0          FB=+3.9755196e+ 0
A=-1.4617200e- 1          B=-2.2544537e- 1
FA=+3.8520386e+ 0          FB=+3.9755196e+ 0
A=-1.4617200e- 1          B=-1.4715601e- 1
FA=+3.8520386e+ 0          FB=+3.8520220e+ 0
A=-1.4617200e- 1          B=-1.4714860e- 1
FA=+3.8520386e+ 0          FB=+3.8520220e+ 0
X
-9.30e- 1  +8.28e- 1
G
-1.72e+ 1  -7.17e+ 0
```



```
H
+2.54e- 2  -5.90e- 2
-5.07e- 2  +1.22e- 1
P
-7.28e- 2  +1.36e- 1
AL=-6.4692453e- 1              FA=+3.6941047e+ 0              GA=+2.1282781e- 1
AL=-1.2938491e+ 0              FA=+3.5811869e+ 0              GA=+1.3124141e- 1
AL=-2.5876981e+ 0              FA=+3.5801218e+ 0              GA=-1.6791936e- 1
A=+0.0000000e+ 0          B=-2.5876981e+ 0
FA=+3.8520220e+ 0          FB=+3.5801218e+ 0
A=-1.9463315e+ 0          B=-2.5876981e+ 0
FA=+3.5327079e+ 0          FB=+3.5901218e+ 0
A=-1.9463315e+ 0          B=-1.9864031e+ 0
FA=+3.5327079e+ 0          FB=+3.5325255e+ 0
A=-1.9463315e+ 0          B=-1.9860771e+ 0
FA=+3.5327079e+ 0          FB=+3.5325256e+ 0
X
-7.85e- 1  +5.58e- 1
G
-2.20e+ 1  -1.18e+ 1
H
+2.45e- 2  -5.73e- 2
-3.86e- 2  +9.97e- 2
P
-5.61e- 2  +9.09e- 2
AL=-7.9854010e- 1              FA=+3.0252867e+ 0              GA=+4.7846132e- 1
AL=-1.5970802e+ 0              FA=+2.7194145e+ 0              GA=+3.1268143e- 1
AL=-3.1941604e+ 0              FA=+2.2850583e+ 0              GA=+2.6433756e- 1
AL=-6.3883208e+ 0              FA=+2.1428899e+ 0              GA=-5.8039900e- 1
A=+0.0000000e+ 0          B=-6.3883208e+ 0
FA=+3.5325254e+ 0          FB=+2.1428899e+ 0
A=-4.3146140e+ 0          B=-6.3883208e+ 0
FA=+2.0015936e+ 0          FB=+2.1428899e+ 0
A=-4.3146140e+ 0          B=-5.3194325e+ 0
FA=+2.0015936e+ 0          FB=+1.8708047e+ 0
A=-5.3172155e+ 0          B=-5.3194325e+ 0
FA=+1.8708039e+ 0          FB=+1.8708047e+ 0
X
-3.27e- 1  +7.40e- 2
G
-6.98e+ 0  -6.61e+ 0
H
+6.45e- 2  -9.95e- 2
-8.19e- 2  +1.45e- 1
P
+9.11e- 2  -2.66e- 1
AL=-3.5534262e- 1              FA=+2.0031989e+ 0              GA=-1.8264626e+ 0
A=+0.0000000e+ 0          B=-3.5534262e- 1
FA=+1.8708039e+ 0          FB=+2.0031989e+ 0
A=+0.0000000e+ 0          B=-1.3177039e- 1
FA=+1.8708039e+ 0          FB=+1.7971421e+ 0
X
-3.39e- 1  +1.09e- 1
G
-3.49e+ 0  -1.19e+ 0
H
+4.00e- 2  -2.80e- 2
-4.40e- 2  +3.49e- 2
P
-8.69e- 2  +5.60e- 2
```

```
AL=-9.6718220e- 1          FA=+1.5859274e+ 0     GA=+1.9633542e- 1
AL=-1.9343644e+ 0          FA=+1.4531318e+ 0     GA=+5.3456482e- 2
AL=-3.8687288e+ 0          FA=+2.1648959e+ 0     GA=-1.0204292e+ 0
A=+0.0000000e+ 0     B=-3.8687288e+ 0
FA=+1.7971421e+ 0     FB=+2.1648959e+ 0
A=-2.0900477e+ 0     B=-3.8687288e+ 0
FA=+1.4478649e+ 0     FB=+2.1648959e+ 0
A=-2.0900477e+ 0     B=-2.1424026e+ 0
FA=+1.4478649e+ 0     FB=+1.4475613e+ 0
A=-2.0900477e+ 0     B=-2.1364780e+ 0
FA=+1.4478649e+ 0     FB=+1.4475563e+ 0
X
-1.54e- 1  -1.06e- 2
G
-4.41e+ 0  -6.83e+ 0
H
+6.37e- 2  -4.33e- 2
-3.06e- 2  +2.62e- 2
P
-7.16e- 2  +1.17e- 2
AL=-1.3788003e+ 0          FA=+1.2007877e+ 0     GA=+1.2851628e- 1
AL=-2.7576005e+ 0          FA=+1.1135471e+ 0     GA=-2.3129740e- 2
A=+0.0000000e+ 0     B=-2.7576005e+ 0
FA=+1.4475562e+ 0     FB=+1.1135471e+ 0
A=-2.5709078e+ 0     B=-2.7576005e+ 0
FA=+1.1120056e+ 0     FB=+1.1135471e+ 0
A=-2.5709078e+ 0     B=-2.6113906e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
```



```
A=-2.5709078e+ 0     B=-2.6113687e+ 0
FA=+1.1120056e+ 0     FB=+1.1118834e+ 0
ul2 limit

X
+3.19e- 2  -4.08e- 2
G
-1.40e+ 0  -8.36e+ 0
H
+4.16e- 2  -3.97e- 2
+6.62e- 4  +2.11e- 2
P
-6.39e- 2  -1.21e- 1
AL=-7.3171730e- 1          FA=+1.0204627e+ 0     GA=-7.9970300e- 1
A=+0.0000000e+ 0     B=-7.3171730e- 1
FA=+1.1119141e+ 0     FB=+1.0204627e+ 0
A=-4.0942476e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
A=-4.0943968e- 1     B=-7.3171730e- 1
FA=+8.9006714e- 1     FB=+1.0204627e+ 0
ul2 limit

X
+6.84e- 2  +2.81e- 2
G
-2.51e+ 0  +4.69e+ 0
```

```
H
+6.71e- 2  +8.46e- 3
-8.08e- 3  +4.60e- 3
P
-2.06e- 1  +3.99e- 4
AL=-4.8549395e- 1          FA=+6.9163861e- 1          GA=+3.3711588e- 1
AL=-9.7098789e- 1          FA=+7.3117810e- 1          GA=-6.8070515e- 1
A=+0.0000000e+ 0       B=-9.7098789e- 1
FA=+9.2305022e- 1       FB=+7.3117810e- 1
A=-6.9528545e- 1       B=-9.7098789e- 1
FA=+6.5015814e- 1       FB=+7.3117810e- 1
A=-6.9528545e- 1       B=-7.1040033e- 1
FA=+6.5015814e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
A=-7.1022385e- 1       B=-7.1040033e- 1
FA=+6.4994063e- 1       FB=+6.4994066e- 1
ul2 limit

X
+2.15e- 1  +2.79e- 2
G
-6.22e- 3  -3.64e+ 0
H
+8.66e- 2  +8.42e- 3
+1.51e- 2  +4.55e- 3
P
-5.55e- 2  -1.66e- 2
AL=-1.7261593e+ 0          FA=+6.3377625e- 1          GA=-6.5191885e- 2
A=+0.0000000e+ 0       B=-1.7261593e+ 0
```



```
FA=+6.4994064e- 1       FB=+6.3377625e- 1
A=-1.0576324e+ 0       B=-1.7261593e+ 0
FA=+6.1360843e- 1       FB=+6.3377625e- 1
A=-1.0576324e+ 0       B=-1.0642330e+ 0
FA=+6.1360843e- 1       FB=+6.1360674e- 1
X
+2.74e- 1  +4.56e- 2
G
+1.76e+ 0  -5.87e+ 0
H
+1.82e- 2  -1.21e- 2
+1.62e- 2  +4.88e- 3
P
-6.30e- 2  -5.00e- 2
AL=-1.2436171e+ 0          FA=+4.4602603e- 1          GA=+9.9916584e- 2
AL=-2.4872341e+ 0          FA=+3.4783017e- 1          GA=+5.8892006e- 2
AL=-4.9744682e+ 0          FA=+4.2336437e- 1          GA=-1.8868479e- 1
A=+0.0000000e+ 0       B=-4.9744682e+ 0
FA=+6.1360673e- 1       FB=+4.2336437e- 1
A=-3.2068384e+ 0       B=-4.9744682e+ 0
FA=+3.1636251e- 1       FB=+4.2336437e- 1
A=-3.2068384e+ 0       B=-3.5970047e+ 0
FA=+3.1636251e- 1       FB=+3.1141123e- 1
A=-3.5741730e+ 0       B=-3.5970047e+ 0
FA=+3.1139396e- 1       FB=+3.1141123e- 1
A=-3.5762786e+ 0       B=-3.5970047e+ 0
FA=+3.1139395e- 1       FB=+3.1141123e- 1
X
+4.99e- 1  +2.24e- 1
G
+3.90e+ 0  -4.91e+ 0
H
+8.64e- 2  +4.21e- 2
+6.43e- 2  +4.31e- 2
P
+2.12e- 2  -4.75e- 2
AL=-1.9210963e+ 0          FA=+1.4128384e+ 0          GA=-1.4403909e+ 0
A=+0.0000000e+ 0       B=-1.9210963e+ 0
FA=+3.1139397e- 1       FB=+1.4128384e+ 0
A=+0.0000000e+ 0       B=-3.3523321e- 1
FA=+3.1139397e- 1       FB=+2.5849476e- 1
X
+4.92e- 1  +2.40e- 1
G
-7.06e- 1  -3.15e- 1
H
+7.32e- 2  +7.18e- 2
+5.66e- 2  +6.02e- 2
P
-6.95e- 2  -6.97e- 2
AL=-1.0159353e+ 0          FA=+1.9422059e- 1          GA=+5.1827330e- 2
AL=-2.0318706e+ 0          FA=+1.7052907e- 1          GA=-1.8624317e- 2
A=+0.0000000e+ 0       B=-2.0318706e+ 0
FA=+2.5849476e- 1       FB=+1.7052907e- 1
A=-1.8234963e+ 0       B=-2.0318706e+ 0
FA=+1.6896358e- 1       FB=+1.7052907e- 1
A=-1.8234963e+ 0       B=-1.8539604e+ 0
FA=+1.6896358e- 1       FB=+1.6891971e- 1
X
+6.21e- 1  +3.69e- 1
```

```
G
+3.17e+ 0  -3.17e+ 0
H
+1.22e- 1  +1.20e- 1
+1.36e- 1  +1.40e- 1
P
-4.43e- 2  -6.03e- 2
AL=-1.3361903e+ 0          FA=+1.1754906e- 1          GA=+2.8426795e- 2
AL=-2.6723807e+ 0          FA=+9.2910534e- 2          GA=+6.6258397e- 3
AL=-5.3447613e+ 0          FA=+2.1192281e- 1          GA=-1.2510027e- 1
A=+0.0000000e+ 0       B=-5.3447613e+ 0
FA=+1.6891969e- 1       FB=+2.1192281e- 1
A=-2.8671343e+ 0       B=-5.3447613e+ 0
FA=+9.2049174e- 2       FB=+2.1192281e- 1
A=-2.8671343e+ 0       B=-2.9691607e+ 0
FA=+9.2049174e- 2       FB=+9.1960642e- 2
A=-2.9526174e+ 0       B=-2.9691607e+ 0
FA=+9.1957127e- 2       FB=+9.1960642e- 2
X
+7.52e- 1  +5.48e- 1
G
+4.73e+ 0  -3.48e+ 0
H
+1.03e- 1  +9.47e- 2
+1.44e- 1  +1.51e- 1
P
-1.55e- 2  -7.75e- 2
AL=-1.2659653e+ 0          FA=+3.1036271e- 1          GA=-5.3704765e- 1
A=+0.0000000e+ 0       B=-1.2659653e+ 0
FA=+9.1957145e- 2       FB=+3.1036271e- 1
A=+0.0000000e+ 0       B=-3.3450603e- 1
FA=+9.1957145e- 2       FB=+5.9199486e- 2
X
+7.57e- 1  +5.73e- 1
G
-6.98e- 1  +1.40e- 1
H
+1.20e- 1  +1.82e- 1
+1.65e- 1  +2.55e- 1
P
-6.09e- 2  -9.14e- 2
AL=-9.1072372e- 1          FA=+3.6207173e- 2          GA=+1.8210381e- 2
AL=-1.8214474e+ 0          FA=+3.4468756e- 2          GA=-2.1101839e- 2
A=+0.0000000e+ 0       B=-1.8214474e+ 0
FA=+5.9199487e- 2       FB=+3.4468756e- 2
A=-1.4122609e+ 0       B=-1.8214474e+ 0
FA=+3.0801743e- 2       FB=+3.4468756e- 2
A=-1.4122609e+ 0       B=-1.4477843e+ 0
FA=+3.0801743e- 2       FB=+3.0773475e- 2
A=-1.4122609e+ 0       B=-1.4475166e+ 0
FA=+3.0801743e- 2       FB=+3.0773481e- 2
X
+8.45e- 1  +7.06e- 1
G
+2.46e+ 0  -1.64e+ 0
H
+1.86e- 1  +2.80e- 1
+2.97e- 1  +4.53e- 1
P
-3.00e- 2  -5.32e- 2
```



```
AL=-1.6382858e+ 0          FA=+1.5364853e- 2          GA=+5.8525769e- 3
AL=-3.2765715e+ 0          FA=+1.2209296e- 2          GA=-2.8956823e- 3
A=+0.0000000e+ 0       B=-3.2765715e+ 0
FA=+3.0773471e- 2       FB=+1.2209296e- 2
A=-2.7570300e+ 0       B=-3.2765715e+ 0
FA=+1.1627267e- 2       FB=+1.2209296e- 2
A=-2.7570300e+ 0       B=-2.8445382e+ 0
FA=+1.1627267e- 2       FB=+1.1604630e- 2
A=-2.8441893e+ 0       B=-2.8445382e+ 0
FA=+1.1604636e- 2       FB=+1.1604630e- 2
A=-2.8441893e+ 0       B=-2.8445319e+ 0
FA=+1.1604636e- 2       FB=+1.1604608e- 2
A=-2.8441893e+ 0       B=-2.8445305e+ 0
FA=+1.1604636e- 2       FB=+1.1604625e- 2
A=-2.8441893e+ 0       B=-2.8445269e+ 0
FA=+1.1604636e- 2       FB=+1.1604617e- 2
A=-2.8441893e+ 0       B=-2.8445249e+ 0
FA=+1.1604636e- 2       FB=+1.1604613e- 2
X
+9.30e- 1  +8.57e- 1
G
+2.91e+ 0  -1.64e+ 0
H
+1.89e- 1  +2.87e- 1
+3.36e- 1  +5.22e- 1
P
+1.10e- 5  -2.23e- 2
AL=-4.4919431e+ 0          FA=+8.4929843e- 1          GA=-4.0951915e- 1
A=+0.0000000e+ 0       B=-4.4919431e+ 0
FA=+1.1604610e- 2       FB=+8.4929843e- 1
A=+0.0000000e+ 0       B=-3.6799590e- 1
FA=+1.1604610e- 2       FB=+4.8807783e- 3
A=+0.0000000e+ 0       B=-3.6799544e- 1
FA=+1.1604610e- 2       FB=+4.8807783e- 3
X
+9.30e- 1  +8.65e- 1
G
-1.40e- 1  -7.00e- 5
H
+1.89e- 1  +3.52e- 1
+3.36e- 1  +6.30e- 1
P
-2.65e- 2  -4.92e- 2
AL=-1.7895508e+ 0          FA=+1.0093632e- 3          GA=+6.8333390e- 5
AL=-3.5791016e+ 0          FA=+8.6562079e- 3          GA=-1.0299320e- 2
A=+0.0000000e+ 0       B=-3.5791016e+ 0
FA=+4.8807783e- 3       FB=+8.6562079e- 3
A=-1.8069452e+ 0       B=-3.5791016e+ 0
FA=+1.0086767e- 3       FB=+8.6562079e- 3
A=-1.8069452e+ 0       B=-1.8105599e+ 0
FA=+1.0086767e- 3       FB=+1.0086576e- 3
A=-1.8069452e+ 0       B=-1.8102758e+ 0
FA=+1.0086767e- 3       FB=+1.0086562e- 3
A=-1.8069452e+ 0       B=-1.8102420e+ 0
FA=+1.0086767e- 3       FB=+1.0086545e- 3
A=-1.8069452e+ 0       B=-1.8102344e+ 0
FA=+1.0086767e- 3       FB=+1.0086565e- 3
X
+9.78e- 1  +9.54e- 1
G
```

```
+8.53e- 1  -4.59e- 1
H
+3.43e- 1  +6.38e- 1
+6.54e- 1  +1.22e+ 0
P
-7.53e- 3  -1.63e- 2
AL=-5.5696592e+ 0              FA=+2.6136984e- 3              GA=-1.1864976e- 3
A=+0.0000000e+ 0        B=-5.5695592e+ 0
FA=+1.0086523e- 3        FB=+2.6136984e- 3
A=+0.0000000e+ 0        B=-1.8800045e+ 0
FA=+1.0086523e- 3        FB=+8.2425441e- 5
A=-1.8628871e+ 0        B=-1.8800045e+ 0
FA=+8.2357047e- 5        FB=+8.2425441e- 5
X*1= +9.92058247e -1 +9.84618723e -1  F*= +8.23575483e -5
```


(vi) The conjugate gradient method of Powell, already implemented in a micro above is included here in a main frame version.

```
PROGRAM powellnd
BEGIN
   [1:4] REAL x0,xmin;
   [1:3] REAL x30,x3min;
   [1:2] REAL x20,x2min;
   [1:10] REAL x100,x10min;
    INT icnt:=0;

    PROC conv=( BOOL bf,bx, REF  BOOL bf1,bx1) BOOL :
    BEGIN
       IF (bf AND bx) OR (bx AND bx1) OR (bf AND bf1) THEN  TRUE
       ELSE
          IF bx THEN bx1:= TRUE
          ELIF bx1 THEN bx1:= FALSE  FI ;
          IF bf THEN bf1:= TRUE
          ELIF bf1 THEN bf1:= FALSE  FI ;
          FALSE
       FI
    END ;

    PROC powell=( PROC ( REF [ ] REAL ) REAL f,
                  REF [ ] REAL d,xk, REAL h,er, REF  REAL fv,f0) REAL :
    BEGIN
       REAL xa,xb,xc,fa,fb,fc,dd,ch,w,max,w1,w2,w3,min;
       INT n:= UPB xk,ul,i;
      [1:n] REAL xar;
       BOOL bfa,bxa,bfb,bxb,bfa1:= FALSE ,bxa1:= FALSE ,
         tsta,tstb,bfb1:= FALSE,bxb1:= FALSE;
      ul:=50;
       BOOL powout:= TRUE ;
       FORMAT fq1=$l"x"3x3(+d.8de+zd3x)l,"f"3(3x+d.8de+zd)$;
      (powout!print((newline,"enter powell line search",newline)));
      max:= ABS d [1];
       FOR i1 FROM 2 TO n DO
          IF max< ABS d [i1] THEN max:= ABS d [i1] FI  OD ;
       FOR i1 TO n DO d[i1]:=d [i1]/max OD ;
      xa:=0.0;

       PROC tr=( REAL ex) VOID :
       BEGIN
          FOR i1 TO n DO xar[i1]:=xk[i1]+ex*d[i1]
       OD  END ;

      tr(xa);
      fa:=f(xar);
      f0:=fa;
      xb:=xa+h;tr(xb);fb:=f(xar);
       IF fb<fa THEN xc:=xb+h;tr(xc);fc:=f(xar)
       ELSE xc:=xa-h;tr(xc);fc:=f(xar);
             w:=xc;xc:=xb;xb:=xa;xa:=w;
             w:=fc;fc:=fb;fb:=fa;fa:=w
       FI ;
       TO ul WHILE
         (powout!printf((fq1,xa,xb,xc,fa,fb,fc)));
```



```
         bfa:= AdS (fa-fb)<er*er*( ABS fa+0.5)*10;
         bxa:= ABS (xa-xb)<er*( ABS xa+0.5);
         bfb:= ABS (fb-fc)<er*er*( ABS fb+0.5)*10;
         bxb:= ABS (xb-xc)<er*( ABS xb+0.5);
         tsta:=conv(bfa,bxa,bfa1,bxa1);
         tstb:=conv(bfb,oxb,bfa1,bxa1);
          NOT tsta AND  NOT tstb
       DO
       BEGIN
         w1:=xb-xc;w2:=xc-xa;w3:=xa-xb;
         w:=(w1*fa+w2*fb+w3*fc);
          IF  ABS w<1.0e-30 THEN print((newline,"fail2",newline))
          ELSE dd:=((xb+xc)*w1*fa+(xc+xa)*w2*fb+(xa+xb)*w3*fc)/w/2.0 FI ;
         ch:=(w1*fa+w2*fb+w3*fc)/w1/w2/w3;
          IF ch>0.0
             OR
            (ch<0.0 AND (dd<xa OR dd>xc) AND
             ( ABS (dd-xa)< ABS (dd-xc)! ABS (dd-xa)! ABS (dd-xc))>h*5)
          THEN
             IF fc<fb THEN
               xa:=xb;fa:=fb;xb:=xc;fb:=fc;
               xc:=xb+h;tr(xc);fc:=f(xar)
             ELSE
               xc:=xb;fc:=fb;xb:=xa;fb:=fa;
               xa:=xb-h;tr(xa);fa:=f(xar)
             FI
          ELSE
             IF dd>xa AND dd<xb AND  ABS (fa-fc)<10*er*( ABS fa+0.5)
             THEN i:=3
             ELIF dd>xb AND dd<xc AND  ABS (fa-fc)<10*er*( ABS fa+0.5)
             THEN i:=1
             ELIF fa>fb AND fa>fc THEN i:=1
             ELIF fb>fa AND fb>fc THEN i:=2
             ELSE i:=3 FI ;
             CASE i IN
               (xa:=dd;tr(xa);fa:=f(xar)),
               (xb:=dd;tr(xb);fb:=f(xar)),
               (xc:=dd/tr(xc);fc:=f(xar))
             ESAC ;
             IF xa>xb THEN w:=xa;xa:=xb;xb:=w;
                           w:=fa;fa:=fb;fb:=w FI ;
             IF xb>xc THEN w:=xb;xb:=xc;xc:=w;
                           w:=fb;fb:=fc;fc:=w FI ;
             IF xa>xb THEN w:=xa;xa:=xb;xb:=w;
                           w:=fa;fa:=fb;fb:=w FI
          FI
       END  OD ;
      i:=1;min:=fa;
       IF min>fb THEN min:=fb;i:=2 FI ;
       IF min>fc THEN i:=3 FI ;
      (powout!print((newline,"leave powell line search",newline)));
       CASE i IN (fv:=fa;xa),(fv:=fb;xb),(fv:=fc;xc) ESAC
    END ;

    PROC npowell=( PROC ( REF [ ] REAL ) REAL f, REF [ ] REAL x0,xmin,
                   REAL h,er) REAL :
    BEGIN
       INT na:= UPB x0,ul,q,m,ilp:=0;
       REAL xm,max,fs,ft,f1,f2,del,f3,w1,w2,fs0,fa1:=0.0,fa2:=0.0,f0,fdum;
       BOOL st:= TRUE ,b1,powcon,go:= TRUE ,bf,bx,bf1:= FALSE ,bx1:= FALSE ;
```

```
[1:na,1:na] REAL de;
[0:na,1:na] REAL y;
[1:na] REAL exk,exk1,exs,af,bff,far;
 FORMAT ft1=$l5(+d.9de+zd5x)$,
        ft2=$l"function calls="4zd5x,"f="+d.9de+zd$;
ul:=300;
 clear( de);
 FOR i1 TO na DO de[i1,i1]:=1.0 OD ;
powcon:= TRUE ;
 clear( exk1);
exk:=x0;
print((newline,newline,"x0="));
printf((ft1,x0));
 WHILE go DO
 BEGIN
 TO ul WHILE
   st OR
   (bx:= TRUE ;
    FOR i1 TO na DO
      bx:=bx AND ( ABS (exk[i1]-exk1[i1])<er*( ABS exk[i1]+0.5))
   OD;
    bf:= ABS (fa1-fa2)<er*er*( ABS fa2+0.5)*10;
    NOT conv(bf,bx,bf1,bx1))
 DO
 BEGIN
   fa1:=fa2;
   exk1:=exk;
   y[0, ]:=exk1;
    FOR i1 TO na DO
    BEGIN
      xm:=powell(f,de[i1, ],y[i1-1, ],h,er,far[i1],fdum);
      (i1=1|f0:=fdum);
       FOR i2 TO na DO y[i1,i2]:=y[i1-1,i2]+xm*de[i1,i2]
   OD  END  OD ;
   m:=1;
   (st|fa1:=f0;st:= FALSE );
   fs:=far[1];
   fs0:=f0;max:=fs0-fs;
    FOR i1 FROM 2 TO na DO
      (ft:=fs;fs:=far[i1];
          IF max<ft-fs THEN max:=ft-fs;m:=i1 FI ) OD ;
   f1:=fs0;f2:=fs;
   del:=(m=1|f0|far[m-1])-far[m];
    FOR i1 TO na DO exs[i1]:=2*y[na,i1]-y[0,i1] OD ;
   f3:=f(exs);
    IF f3>=f1 OR
      (w1:=f1-f2-del;w2:=f1-f3;
      (f1-2*f2+f3)*w1*w1>=del*w2*w2/2)
    THEN
      exk:=y[na, ];fa2:=far[na]
    ELSE
      FOR i1 TO na DO exs[i1]:=y[na,i1]-exk1[i1] OD ;
      xm:=powell(f,exs,y[na, ],h,er,fa2,fdum);
       FOR i1 TO na DO exk[i1]:=y[na,i1]+xm*exs[i1] OD ;
       FOR i1 FROM m+1 TO na DO de[i1-1, ]:=de[i1, ] OD ;
      de[na, ]:=exs
    FI
   ;print((newline,"exk="));
   printf((ft1,exk));
   printf((ft2,icnt,fa2))
```



```
   END  OD ;
   IF powcon THEN
      IF ilp=0 THEN
        print((newline,"ilp=0",newline));
        af:=exk;
         FOR i1 TO na DO exk[i1]:=exk[i1]+40.0*er OD ;
        ilp:=1;
        st:= TRUE
     ELIF ilp=1 THEN
        print((newline,"ilp=1",newline));
        bff:=exk;
         FOR i1 TO na DO exs[i1]:=bff[i1]-af[i1] OD ;
        xm:=powell(f,exs,af,h,er,fa2,fdum);
         FOR i1 TO na DO exs[i1]:=af[i1]+xm*exs[i1] OD ;
         IF
           b1:= TRUE ;
            FOR i1 TO na DO
              b1:=b1 AND  ABS (af[i1]-exs[i1])<er*( ABS af[i1]+0.5)
           OD;
            FOR i1 TO na DO
              b1:=b1 AND  ABS (bff[i1]-exs[i1])<er*( ABS bff[i1]+0.5)
           OD;
           b1 OR  ABS xm<er/10
         THEN go:= FALSE ;xmin:=exs
         ELSE
           ilp:=0;st:= TRUE ;
            FOR i1 TO na DO de[na,i1]:=af[i1]-exs[i1]
         OD  FI
     FI
   ELSE go:= FALSE ;xmin:=exk
   FI
   END  OD ;
  f(exk)
 END ;

 PROC f=( REF [ ] REAL x) REAL :
 BEGIN
    REAL f1,f2,f3,f4;
   icnt PLUSAB 1;
   f1:=0.2*x[1]+0.32*x[2]+0.12*x[3]+0.30*x[4]-0.94;
   f2:=0.1*x[1]+0.15*x[2]+0.24*x[3]+0.32*x[4]-0.81;
   f3:=0.2*x[1]+0.24*x[2]+0.46*x[3]+0.36*x[4]-1.26;
   f4:=0.6*x[1]+0.40*x[2]+0.32*x[3]+0.20*x[4]-1.52;
   f1*f1+f2*f2+f3*f3+f4*f4
 END ;


 PROC f1=( REF [ ] REAL x) REAL :
 BEGIN
   REAL w:=(x[2]-x[1]*x[1]);
  icnt PLUSAB 1;
  100*w*w+(1-x[1])*(1-x[1])
 END ;

 PROC f2=( REF [ ] REAL x) REAL :
 BEGIN
   REAL w1:=x[1]+10*x[2],
        w2:=x[3]-x[4],
        w3:=x[2]-2*x[3],
        w4:=x[1]-x[4];
```

```
    icnt PLUSAB 1;
    w1*w1+5*w2*w2+w3*w3*w3*w3+w4*w4*w4*w4+10
  END ;

  PROC f3=( REF [ ] REAL x) REAL :
  BEGIN
    icnt PLUSAB 1;
    x[1]*x[1]*x[1]*(x[1]-x[2])-x[1]*x[2]*x[3]*(x[1]-x[4])
  END ;

  PROC f4=( REF [ ] REAL x) REAL :
  BEGIN
     REAL w1:=x[2]-1,w2:=x[4]-1,w3:=x[1]*x[1]-x[2],w4:=x[1]-[illegible]
          w5:=x[3]-1,w6:=x[3]*x[3]-x[4];
    icnt PLUSAB 1;
    100*w3*w3+w4*w4+w5*w5+90*w6*w6+10.1*(w1*w1+w2*w2)+19.8*w[illegible]
  END ;

  PROC f5=( REF [ ] REAL x) REAL :
  BEGIN
     REAL w1:=exp(x[1])-x[2],w2:=x[2]-x[3],w3:=tan(x[3]-x[4]);
     REAL v1:=w1*w1,v2:=w2*w2*w2,v3:=w3*w3,v4:=x[1]*x[1];
    icnt PLUSAB 1;
    v1*v1+100*v2*v2+v3*v3+v4*v4*v4*v4
  END ;

  PROC f6=( REF [ ] REAL x) REAL :
  BEGIN
     REAL r:=sqrt(x[1]*x[1]+x[2]*x[2]),th;
    icnt PLUSAB 1;
    th:=( ABS (x[1]-x[2])<1.0e-30!
           (x[1]>0!0.125!0.625)
      !: ABS x[1]<1.0e-30!
           (x[1]>0!0.25!0.75)
     !(x[1]>0.0!arctan(x[2]/x[1])/2/pi
                !arctan(x[2]/x[1])/2/pi+0.5));
    100*((x[3]-10*th)*(x[3]-10*th)+(r-1)*(r-1))+x[3]*x[3]
  END ;

  PROC f7=( REF [ ] REAL x) REAL :
  BEGIN
     REAL s,w;
    w:=1-x[1];
    icnt PLUSAB 1;
    s:=0.0;
    s PLUSAB w*w;
    w:=1-x[10];
    s PLUSAB w*w;
     FOR i1 TO 9 DO (w:=x[i1]*x[i1]-x[i1+1];s PLUSAB w*w) OD;
    s
  END ;


  GOTO la;
 x0:=(0.0,0.0,0.0,0.0);
 npowell(f,x0,xmin,0.5,1.0e-3);
la:SKIP;
 x20:=(-1.2,1.0);
 npowell(f1,x20,x2min,0.5,1.0e-03);
 GOTO lb;
```



```
   x0:=(3.0,-1.0,0.0,1.0);
   npowell(f2,x0,xmin,0.5,1.0e-03);
   x0:=(0.0,-1.0,-2.0,-3.0);
   npowell(f3,x0,xmin,0.5,1.0e-03);
   x0:=(-3.0,-1.0,-3.0,-1.0);
   npowell(f4,x0,xmin,0.5,1.0e-03);
   x0:=(1.0,2.0,2.0,2.0);
   npowell(f5,x0,xmin,0.5,1.0e-03);
   x30:=(-1.0,0.0,0.0);
   npowell(f6,x30,x3min,0.5,1.0e-03);
    FOR i1 TO 10 DO x100[i1]:=-2.0 OD ;
   npowell(f7,x100,x10min,0.5,1.0e-03);
lb: SKIP ;
    SKIP
 END
 FINISH
```

Powell conjugate gradients.

```
x0=
-1.200000003e+ 0     +1.000000000e+ 0
enter powell line search

x   -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +3.64500008e+ 2   +2.42000012e+ 1   +2.88999999e+ 1
x   +0.00000000e+ 0   +2.43188405e- 1   +5.00000000e- 1
f   +2.42000012e+ 1   +4.54333210e+ 0   +2.88999999e+ 1
x   +0.00000000e+ 0   +2.36622764e- 1   +2.43188405e- 1
f   +2.42000012e+ 1   +4.37187302e+ 0   +4.54333210e+ 0
x   +2.11015077e- 1   +2.36622764e- 1   +2.43188405e- 1
f   +4.00406063e+ 0   +4.37187302e+ 0   +4.54333210e+ 0
x   +2.04156404e- 1   +2.11015077e- 1   +2.36622764e- 1
f   +3.99027327e+ 0   +4.00406063e+ 0   +4.37187302e+ 0
x   +2.04156404e- 1   +2.04944219e- 1   +2.11015077e- 1
f   +3.99027327e+ 0   +3.98997742e+ 0   +4.00406063e+ 0
x   +2.04944219e- 1   +2.05027096e- 1   +2.11015077e- 1
f   +3.98997742e+ 0   +3.98997483e+ 0   +4.00406063e+ 0
leave powell line search

enter powell line search

x   -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +2.79870834e+ 1   +3.98997483e+ 0   +2.99928660e+ 1
x   -5.00000000e- 1   -1.00289130e- 2   +0.00000000e+ 0
f   +2.79870834e+ 1   +3.97991690e+ 0   +3.98997483e+ 0
x   -1.00289250e- 2   -1.00289130e- 2   +0.00000000e+ 0
f   +3.97991690e+ 0   +3.97991690e+ 0   +3.98997483e+ 0
leave powell line search

enter powell line search

x   -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +1.55190666e+ 2   +3.97991690e+ 0   +5.41491776e+ 1
x   +0.00000000e+ 0   +1.25436343e- 1   +5.00000000e- 1
f   +3.97991690e+ 0   +8.68178582e+ 0   +5.41491776e+ 1
x   -4.89699449e- 2   +0.00000000e+ 0   +1.25436343e- 1
f   +5.12735724e+ 0   +3.97991690e+ 0   +8.68178582e+ 0
x   -4.89699449e- 2   +0.00000000e+ 0   +9.05818469e- 3
f   +5.12735724e+ 0   +3.97991690e+ 0   +3.97448280e+ 0
x   +0.00000000e+ 0   +5.29145019e- 3   +9.05818469e- 3
```



```
f   +3.97991690e+ 0   +3.96932393e+ 0   +3.97448280e+ 0
x   +5.29145019e- 3   +5.33498073e- 3   +9.05818469e- 3
f   +3.96932393e+ 0   +3.96932355e+ 0   +3.97448280e+ 0
leave powell line search

exk=
-9.896379262e- 1     +9.897101074e- 1
function calls=   23     f=+3.969323546e+ 0
enter powell line search.

x   -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +2.79366353e+ 1   +3.96932355e+ 0   +3.00020120e+ 1
x   -5.00000000e- 1   -1.03268836e- 2   +0.00000000e+ 0
f   +2.79366353e+ 1   +3.95865908e+ 0   +3.96932355e+ 0
x   -1.03268836e- 2   -1.03268781e- 2   +0.00000000e+ 0
f   +3.95865908e+ 0   +3.95865908e+ 0   +3.96932355e+ 0
leave powell line search

enter powell line search

x   -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +1.53864626e+ 2   +3.95865908e+ 0   +5.33673162e+ 1
x   +0.00000000e+ 0   +1.26053607e- 1   +5.00000000e- 1
f   +3.95865908e+ 0   +8.64581537e+ 0   +5.33673162e+ 1
x   -4.97751995e- 2   +0.00000000e+ 0   +1.26053607e- 1
f   +5.13064611e+ 0   +3.95865908e+ 0   +8.64581537e+ 0
x   -4.97751995e- 2   +0.00000000e+ 0   +9.19797213e- 3
f   +5.13064611e+ 0   +3.95865908e+ 0   +3.95336816e+ 0
x   +0.00000000e+ 0   +5.33739210e- 3   +9.19797213e- 3
f   +3.95865908e+ 0   +3.94800538e+ 0   +3.95336816e+ 0
x   +5.33739210e- 3   +5.38048346e- 3   +9.19797213e- 3
f   +3.94800538e+ 0   +3.94800502e+ 0   +3.95336816e+ 0
leave powell line search

enter powell line search

x   +0.00000000e+ 0   +5.00000000e- 1   +1.00000000e+ 0
f   +3.94800502e+ 0   +3.28641602e+ 0   +8.31148136e+ 0
x   +0.00000000e+ 0   +3.08170322e- 1   +5.00000000e- 1
f   +3.94800502e+ 0   +3.36035591e+ 0   +3.28641602e+ 0
x   +3.08170322e- 1   +4.67420310e- 1   +5.00000000e- 1
f   +3.36035591e+ 0   +3.26259106e+ 0   +3.28641602e+ 0
x   +4.31568474e- 1   +4.67420310e- 1   +5.00000000e- 1
f   +3.25701904e+ 0   +3.26259106e+ 0   +3.28641602e+ 0
x   +4.31568474e- 1   +4.40260671e- 1   +4.67420310e- 1
f   +3.25701904e+ 0   +3.25655618e+ 0   +3.26259106e+ 0
x   +4.31568474e- 1   +4.39382318e- 1   +4.40260671e- 1
f   +3.25701904e+ 0   +3.25655240e+ 0   +3.25655618e+ 0
leave powell line search

exk=
-7.610210851e- 1     +5.397377238e- 1
function calls=   45     f=+3.256552398e+ 0
enter powell line search

x   -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +1.10375488e+ 2   +3.25655240e+ 0   +2.15843213e+ 1
x   +0.00000000e+ 0   +1.76949978e- 1   +5.00000000e- 1
f   +3.25655240e+ 0   +6.11712062e+ 0   +2.15843213e+ 1
x   -3.89659037e- 2   +0.00000000e+ 0   +1.76949978e- 1
```

```
f    +4.20693898e+ 0    +3.25655240e+ 0    +6.11712062e+ 0
x    -3.89659037e- 2    +0.00000000e+ 0    +4.54422166e- 2
f    +4.20693898e+ 0    +3.25655240e+ 0    +3.00804114e+ 0
x    +0.00000000e+ 0    +3.49190277e- 2    +4.54422166e- 2
f    +3.25655240e+ 0    +2.99110410e+ 0    +3.00804114e+ 0
x    +3.49190277e- 2    +3.62105560e- 2    +4.54422166e- 2
f    +2.99110410e+ 0    +2.99088818e+ 0    +3.00804114e+ 0
x    +3.49190277e- 2    +3.59992189e- 2    +3.62105560e- 2
f    +2.99110410e+ 0    +2.99087897e+ 0    +2.99088818e+ 0
leave powell line search

enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +4.54750955e+ 0    +2.99087897e+ 0    +5.54397768e+ 0
x    -5.00000000e- 1    -6.06164089e- 2    +0.00000000e+ 0
f    +4.54750955e+ 0    +3.15759724e+ 0    +2.99087897e+ 0
x    -6.06164089e- 2    +0.00000000e+ 0    +1.63480927e+ 0
f    +3.15759724e+ 0    +2.99087897e+ 0    +1.23561567e+ 2
x    -6.06164089e- 2    +0.00000000e+ 0    +1.68362823e- 4
f    +3.15759724e+ 0    +2.99087897e+ 0    +2.99047482e+ 0
x    +0.00000000e+ 0    +1.68362823e- 4    +2.08537161e- 1
f    +2.99087897e+ 0    +2.99047482e+ 0    +2.91092926e+ 0
leave powell line search

exk=
-6.190706864e- 1      +3.294396549e- 1
function calls=   62      f=+2.910929263e+ 0
enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +8.52066689e+ 1    +2.91092926e+ 0    +9.70903075e+ 0
x    +0.00000000e+ 0    +2.11848645e- 1    +5.00000000e- 1
f    +2.91092926e+ 0    +4.32874602e+ 0    +9.70903075e+ 0
x    -3.37475087e- 2    +0.00000000e+ 0    +2.11848645e- 1
f    +3.63585103e+ 0    +2.91092926e+ 0    +4.32874602e+ 0
x    -3.37475087e- 2    +0.00000000e+ 0    +7.67535698e- 2
f    +3.63585103e+ 0    +2.91092926e+ 0    +2.47845525e+ 0
x    +0.00000000e+ 0    +5.80227668e- 2    +7.67535698e- 2
f    +2.91092926e+ 0    +2.45085770e+ 0    +2.47845525e+ 0
x    +5.80227668e- 2    +6.13745204e- 2    +7.67535698e- 2
f    +2.45085770e+ 0    +2.45017186e+ 0    +2.47845525e+ 0
x    +5.80227668e- 2    +6.06363937e- 2    +6.13745204e- 2
f    +2.45085770e+ 0    +2.45010579e+ 0    +2.45017186e+ 0
x    +5.80227668e- 2    +6.06076610e- 2    +6.06363937e- 2
f    +2.45085770e+ 0    +2.45010570e+ 0    +2.45010579e+ 0
leave powell line search

enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +6.05148453e+ 0    +2.45010570e+ 0    +8.78787541e+ 0
x    -5.00000000e- 1    -6.88286042e- 2    +0.00000000e+ 0
f    +6.05148453e+ 0    +2.72514483e+ 0    +2.45010570e+ 0
x    -6.88286042e- 2    +0.00000000e+ 0    +2.34231098e- 1
f    +2.72514483e+ 0    +2.45010570e+ 0    +3.08969572e+ 0
x    -6.88286042e- 2    +0.00000000e+ 0    +5.56035195e- 2
f    +2.72514483e+ 0    +2.45010570e+ 0    +2.35206494e+ 0
x    +0.00000000e+ 0    +5.56035195e- 2    +7.69331921e- 2
f    +2.45010570e+ 0    +2.35206494e+ 0    +2.34934390e+ 0
```



```
x    +5.56035195e- 2    +6.92685442e- 2    +7.69331921e- 2
f    +2.35206494e+ 0    +2.34792313e+ 0    +2.34934390e+ 0
x    +5.56035195e- 2    +6.90534571e- 2    +6.92685442e- 2
f    +2.35206494e+ 0    +2.34792265e+ 0    +2.34792313e+ 0
leave powell line search

enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +2.11989071e+ 1    +2.34792265e+ 0    +2.47361603e+ 0
x    +0.00000000e+ 0    +2.46688213e- 1    +5.00000000e- 1
f    +2.34792265e+ 0    +1.63223933e+ 0    +2.47361603e+ 0
x    +0.00000000e+ 0    +2.39900369e- 1    +2.46688213e- 1
f    +2.34792265e+ 0    +1.64853521e+ 0    +1.63223933e+ 0
x    +2.39900369e- 1    +2.46688213e- 1    +8.18748124e- 1
f    +1.64853521e+ 0    +1.63223933e+ 0    +2.03395714e+ 1
x    +2.39900369e- 1    +2.46688213e- 1    +2.63039570e- 1
f    +1.64853521e+ 0    +1.63223933e+ 0    +1.59537873e+ 0
x    +2.46688213e- 1    +2.63039570e- 1    +4.32942837e- 1
f    +1.63223933e+ 0    +1.59537873e+ 0    +1.77569631e+ 0
x    +2.46688213e- 1    +2.63039570e- 1    +3.18181757e- 1
f    +1.63223933e+ 0    +1.59537873e+ 0    +1.51082325e+ 0
x    +2.63039570e- 1    +3.18181757e- 1    +3.66649579e- 1
f    +1.59537873e+ 0    +1.51082325e+ 0    +1.52365881e+ 0
x    +3.18181757e- 1    +3.34786337e- 1    +3.66649579e- 1
f    +1.51082325e+ 0    +1.50340231e+ 0    +1.52365881e+ 0
x    +3.34786337e- 1    +3.36487893e- 1    +3.66649579e- 1
f    +1.50340231e+ 0    +1.50326551e+ 0    +1.52365881e+ 0
x    +3.34786337e- 1    +3.36487893e- 1    +3.37328993e- 1
f    +1.50340231e+ 0    +1.50326551e+ 0    +1.50324373e+ 0
x    +3.36487893e- 1    +3.37328993e- 1    +3.37452140e- 1
f    +1.50326551e+ 0    +1.50324373e+ 0    +1.50324310e+ 0
leave powell line search

exk=
-1.859269924e- 1      +3.452890494e- 3
function calls=   96      f=+1.503243104e+ 0
enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +1.16764808e+ 1    +1.50324310e+ 0    +2.59871361e+ 1
x    -5.00000000e- 1    -1.03230238e- 1    +0.00000000e+ 0
f    +1.16764808e+ 1    +1.78217939e+ 0    +1.50324310e+ 0
x    -1.03230238e- 1    -2.12342730e- 2    +0.00000000e+ 0
f    +1.78217939e+ 0    +1.45175536e+ 0    +1.50324310e+ 0
x    -3.00072480e- 2    -2.12342730e- 2    +0.00000000e+ 0
f    +1.44773042e+ 0    +1.45175536e+ 0    +1.50324310e+ 0
x    -3.00072480e- 2    -2.91220977e- 2    -2.12342730e- 2
f    +1.44773042e+ 0    +1.44768749e+ 0    +1.45175536e+ 0
x    -3.00072480e- 2    -2.91874113e- 2    -2.91220977e- 2
f    +1.44773042e+ 0    +1.44768724e+ 0    +1.44768749e+ 0
leave powell line search

enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +3.56687039e+ 0    +1.44768724e+ 0    +1.92581329e+ 1
x    -5.00000000e- 1    -1.96833352e- 1    +0.00000000e+ 0
f    +3.56687039e+ 0    +2.00478968e+ 0    +1.44768724e+ 0
x    -1.96833352e- 1    +0.00000000e+ 0    +2.06283385e- 1
```

```
f    +2.00478968e+ 0   +1.44768724e+ 0   +2.49304461e+ 0
x    -1.96833352e- 1   -2.61853931e- 2   +0.00000000e+ 0
f    +2.00478968e+ 0   +1.50545710e+ 0   +1.44768724e+ 0
x    -2.61853931e- 2   +0.00000000e+ 0   +2.88507894e- 1
f    +1.50545710e+ 0   +1.44768724e+ 0   +4.52601099e+ 0
x    -2.61853931e- 2   +0.00000000e+ 0   +1.38672478e- 2
f    +1.50545710e+ 0   +1.44768724e+ 0   +1.42489478e+ 0
x    +0.00000000e+ 0   +1.38672478e- 2   +6.54431889e- 2
f    +1.44768724e+ 0   +1.42489478e+ 0   +1.41089220e+ 0
x    +1.38672478e- 2   +4.61296653e- 2   +6.54431889e- 2
f    +1.42489478e+ 0   +1.40069251e+ 0   +1.41089220e+ 0
x    +1.38672478e- 2   +4.51323185e- 2   +4.61296653e- 2
f    +1.42489478e+ 0   +1.40072313e+ 0   +1.40069251e+ 0
x    +4.51323185e- 2   +4.61296653e- 2   +4.62988634e- 2
f    +1.40072313e+ 0   +1.40069251e+ 0   +1.40069240e+ 0
leave powell line search

enter powell line search

x    -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f    +1.43402524e+ 1   +1.40069240e+ 0   +4.90116227e+ 0
x    +0.00000000e+ 0   +1.43538214e- 1   +5.00000000e- 1
f    +1.40069240e+ 0   +1.10107671e+ 0   +4.90116227e+ 0
x    +0.00000000e+ 0   +1.12704355e- 1   +1.43538214e- 1
f    +1.40069240e+ 0   +1.14385737e+ 0   +1.10107671e+ 0
x    +1.12704355e- 1   +1.43538214e- 1   +2.39831710e- 1
f    +1.14385737e+ 0   +1.10107671e+ 0   +1.11330788e+ 0
x    +1.43538214e- 1   +1.86353879e- 1   +2.39831710e- 1
f    +1.10107671e+ 0   +1.07208535e+ 0   +1.11330788e+ 0
x    +1.86353879e- 1   +1.87461354e- 1   +2.39831710e- 1
f    +1.07208535e+ 0   +1.07193981e+ 0   +1.11330788e+ 0
x    +1.86353879e- 1   +1.87461354e- 1   +1.90720389e- 1
f    +1.07208535e+ 0   +1.07193981e+ 0   +1.07171853e+ 0
x    +1.87461354e- 1   +1.90720389e- 1   +1.91421434e- 1
f    +1.07193981e+ 0   +1.07171853e+ 0   +1.07171200e+ 0
x    +1.90720389e- 1   +1.91398611e- 1   +1.91421434e- 1
f    +1.07171853e+ 0   +1.07171199e+ 0   +1.07171200e+ 0
leave powell line search

exk=
+3.694127500e- 2     -3.661295120e- 2
function calls=  128      f=+1.071711987e+ 0
enter powell line search

x    -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f    +3.70972103e+ 0   +1.07171199e+ 0   +4.93856726e+ 1
x    -5.00000000e- 1   -2.24112786e- 1   +0.00000000e+ 0
f    +3.70972103e+ 0   +2.35071501e+ 0   +1.07171199e+ 0
x    -2.24112786e- 1   +0.00000000e+ 0   +5.00000000e- 1
f    +2.35071501e+ 0   +1.07171199e+ 0   +4.93856726e+ 1
x    -2.24112786e- 1   -9.18654101e- 2   +0.00000000e+ 0
f    +2.35071501e+ 0   +1.19994241e+ 0   +1.07171199e+ 0
x    -9.18654101e- 2   -2.45231837e- 2   +0.00000000e+ 0
f    +1.19994241e+ 0   +1.00884676e+ 0   +1.07171199e+ 0
x    -3.40621141e- 2   -2.45231837e- 2   +0.00000000e+ 0
f    +1.00631891e+ 0   +1.00884676e+ 0   +1.07171199e+ 0
x    -3.40621141e- 2   -3.12562282e- 2   -2.45231837e- 2
f    +1.00631891e+ 0   +1.00587994e+ 0   +1.00884676e+ 0
x    -3.40621141e- 2   -3.14095207e- 2   -3.12562282e- 2
f    +1.00631891e+ 0   +1.00587891e+ 0   +1.00587994e+ 0
```



```
leave powell line search

enter powell line search

x    -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f    +5.04194075e+ 0   +1.00587891e+ 0   +1.30912125e+ 1
x    -5.00000000e- 1   -1.24822814e- 1   +0.00000000e+ 0
f    +5.04194075e+ 0   +1.25508247e+ 0   +1.00587891e+ 0
x    -1.24822814e- 1   -7.69952458e- 4   +0.00000000e+ 0
f    +1.25508247e+ 0   +1.00703239e+ 0   +1.00587891e+ 0
x    -7.69952458e- 4   +0.00000000e+ 0   +1.86082950e- 1
f    +1.00703239e+ 0   +1.00587891e+ 0   +1.34476176e+ 0
x    -7.69952458e- 4   +0.00000000e+ 0   +4.17823670e- 2
f    +1.00703239e+ 0   +1.00587891e+ 0   +9.59248111e- 1
x    +0.00000000e+ 0   +4.17823670e- 2   +8.30377843e- 2
f    +1.00587891e+ 0   +9.59248111e- 1   +9.58486788e- 1
x    +4.17823670e- 2   +6.31081434e- 2   +8.30377843e- 2
f    +9.59248111e- 1   +9.51708972e- 1   +9.58486788e- 1
x    +4.17823670e- 2   +6.29589222e- 2   +6.31081434e- 2
f    +9.59248111e- 1   +9.51712608e- 1   +9.51708972e- 1
leave powell line search

enter powell line search

x    -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f    +1.83952923e+ 1   +9.51708972e- 1   +2.55007187e+ 0
x    +0.00000000e+ 0   +2.08030472e- 1   +5.00000000e- 1
f    +9.51708972e- 1   +5.63945800e- 1   +2.55007187e+ 0
x    +0.00000000e+ 0   +1.57784818e- 1   +2.08030472e- 1
f    +9.51708972e- 1   +6.19134083e- 1   +5.63945800e- 1
x    +1.57784818e- 1   +2.08030472e- 1   +2.96090223e- 1
f    +6.19134083e- 1   +5.63945800e- 1   +5.84304139e- 1
x    +2.08030472e- 1   +2.40035864e- 1   +2.96090223e- 1
f    +5.63945800e- 1   +5.48183836e- 1   +5.84304139e- 1
x    +2.08030472e- 1   +2.40035864e- 1   +2.43106600e- 1
f    +5.63945800e- 1   +5.48183836e- 1   +5.47784656e- 1
x    +2.40035864e- 1   +2.43106600e- 1   +2.47861071e- 1
f    +5.48183836e- 1   +5.47784656e- 1   +5.47615997e- 1
x    +2.43106600e- 1   +2.46947924e- 1   +2.47861071e- 1
f    +5.47784656e- 1   +5.47604866e- 1   +5.47615997e- 1
x    +2.43106600e- 1   +2.46912228e- 1   +2.46947924e- 1
f    +5.47784656e- 1   +5.47604866e- 1   +5.47604866e- 1
leave powell line search

exk=
+3.155521266e- 1     +7.144202851e- 2
function calls=  160      f=+5.476048663e- 1
enter powell line search

x    -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f    +3.02364528e+ 0   +5.47604866e- 1   +4.67608261e+ 1
x    -5.00000000e- 1   -2.24573035e- 1   +0.00000000e+ 0
f    +3.02364528e+ 0   +1.89831309e+ 0   +5.47604866e- 1
x    -2.24573035e- 1   +0.00000000e+ 0   +5.00000000e- 1
f    +1.89831309e+ 0   +5.47604866e- 1   +4.67608261e+ 1
x    -2.24573035e- 1   -9.01514897e- 2   +0.00000000e+ 0
f    +1.89831309e+ 0   +7.35748105e- 1   +5.47604866e- 1
x    -9.01514897e- 2   -9.36260307e- 3   +0.00000000e+ 0
f    +7.35748105e- 1   +5.23919657e- 1   +5.47604866e- 1
x    -2.68156761e- 2   -9.36260307e- 3   +0.00000000e+ 0
```

```
f    +5.10945305e- 1    +5.23919657e- 1    +5.47604866e- 1
x    -2.68156761e- 2    -2.36686741e- 2    -9.36260307e- 3
f    +5.10945305e- 1    +5.10451987e- 1    +5.23919657e- 1
x    -2.68156761e- 2    -2.39964935e- 2    -2.36686741e- 2
f    +5.10945305e- 1    +5.10447308e- 1    +5.10451987e- 1
leave powell line search

enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +4.07213259e+ 0    +5.10447308e- 1    +1.27802795e+ 1
x    -5.00000000e- 1    -1.37512827e- 1    +0.00000000e+ 0
f    +4.07213259e+ 0    +7.15814002e- 1    +5.10447308e- 1
x    -1.37512827e- 1    -2.06784420e- 2    +0.00000000e+ 0
f    +7.15814002e- 1    +5.34799121e- 1    +5.10447308e- 1
x    -2.06784420e- 2    +0.00000000e+ 0    +2.07507072e- 1
f    +5.34799121e- 1    +5.10447308e- 1    +1.11266345e+ 0
x    -2.06784420e- 2    +0.00000000e+ 0    +2.25939639e- 2
f    +5.34799121e- 1    +5.10447308e- 1    +4.90473285e- 1
x    +0.00000000e+ 0    +2.25939639e- 2    +7.64444945e- 2
f    +5.10447308e- 1    +4.90473285e- 1    +4.89545643e- 1
x    +2.25939639e- 2    +5.02788210e- 2    +7.64444945e- 2
f    +4.90473285e- 1    +4.79816093e- 1    +4.89545643e- 1
x    +2.25939639e- 2    +5.01321028e- 2    +5.02788210e- 2
f    +4.90473285e- 1    +4.79822055e- 1    +4.79816083e- 1
leave powell line search

enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +1.59948586e+ 1    +4.79816083e- 1    +2.59753919e+ 0
x    +0.00000000e+ 0    +1.89949207e- 1    +5.00000000e- 1
f    +4.79816083e- 1    +2.43033165e- 1    +2.59753919e+ 0
x    +0.00000000e+ 0    +1.30225996e- 1    +1.89949207e- 1
f    +4.79816083e- 1    +2.87917811e- 1    +2.43033165e- 1
x    +1.30225996e- 1    +1.89949207e- 1    +2.58943874e- 1
f    +2.87917811e- 1    +2.43033165e- 1    +2.55823225e- 1
x    +1.89949207e- 1    +2.11712616e- 1    +2.58943874e- 1
f    +2.43033165e- 1    +2.36687293e- 1    +2.55823225e- 1
x    +1.89949207e- 1    +2.11712616e- 1    +2.15268010e- 1
f    +2.43033165e- 1    +2.36687293e- 1    +2.36390170e- 1
x    +2.11712616e- 1    +2.15268010e- 1    +2.18576470e- 1
f    +2.36687293e- 1    +2.36390170e- 1    +2.36325642e- 1
x    +2.15268010e- 1    +2.17969820e- 1    +2.18576470e- 1
f    +2.36390170e- 1    +2.36321753e- 1    +2.36325642e- 1
leave powell line search

exk=
+5.598042756e- 1      +2.927533016e- 1
function calls=  191      f=+2.363217529e- 1
enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +1.79140249e+ 0    +2.36321753e- 1    +4.05186353e+ 1
x    -5.00000000e- 1    -2.31415182e- 1    +0.00000000e+ 0
f    +1.79140249e+ 0    +1.35666177e+ 0    +2.36321753e- 1
x    -2.31415182e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +1.35666177e+ 0    +2.36321753e- 1    +4.05186353e+ 1
x    -2.31415182e- 1    -9.49773584e- 2    +0.00000000e+ 0
f    +1.35666177e+ 0    +4.45242066e- 1    +2.36321753e- 1
```



```
x    -9.49773584e- 2    +0.00000000e+ 0    +9.31850250e- 3
f    +4.45242066e- 1    +2.36321753e- 1    +2.61463955e- 1
x    -2.40680589e- 2    +0.00000000e+ 0    +9.31850250e- 3
f    +2.16832848e- 1    +2.36321753e- 1    +2.61463955e- 1
x    -2.40680589e- 2    -1.91922460e- 2    +0.00000000e+ 0
f    +2.16832848e- 1    +2.15869619e- 1    +2.36321753e- 1
x    -2.40680589e- 2    -1.97481362e- 2    -1.91922460e- 2
f    +2.16832848e- 1    +2.15860045e- 1    +2.15869619e- 1
x    -1.97481362e- 2    -1.96432970e- 2    -1.91922460e- 2
f    +2.15860045e- 1    +2.15859471e- 1    +2.15869619e- 1
leave powell line search

enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +3.79487541e+ 0    +2.15859471e- 1    +1.18188008e+ 1
x    -5.00000000e- 1    -1.32129299e- 1    +0.00000000e+ 0
f    +3.79487541e+ 0    +3.50541890e- 1    +2.15859471e- 1
x    -1.32129299e- 1    -3.55224763e- 2    +0.00000000e+ 0
f    +3.50541890e- 1    +2.45675826e- 1    +2.15859471e- 1
x    -3.55224763e- 2    +0.00000000e+ 0    +2.07538301e- 1
f    +2.45675826e- 1    +2.15859471e- 1    +8.01082402e- 1
x    -3.55224763e- 2    +0.00000000e+ 0    +1.01160436e- 2
f    +2.45675826e- 1    +2.15859471e- 1    +2.09482443e- 1
x    +0.00000000e+ 0    +1.01160436e- 2    +7.38927899e- 2
f    +2.15859471e- 1    +2.09482443e- 1    +2.11327113e- 1
x    +1.01160436e- 2    +4.03835950e- 2    +7.38927899e- 2
f    +2.09482443e- 1    +1.99371442e- 1    +2.11327113e- 1
x    +4.03835950e- 2    +4.06693295e- 2    +7.38927899e- 2
f    +1.99371442e- 1    +1.99355654e- 1    +2.11327113e- 1
x    +4.03835950e- 2    +4.06693295e- 2    +4.27531246e- 2
f    +1.99371442e- 1    +1.99355654e- 1    +1.99293412e- 1
x    +4.06693295e- 2    +4.27531246e- 2    +4.31109597e- 2
f    +1.99355664e- 1    +1.99293412e- 1    +1.99292207e- 1
leave powell line search

enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +5.46153229e+ 0    +1.99292207e- 1    +8.22515778e- 1
x    +0.00000000e+ 0    +1.97053995e- 1    +5.00000000e- 1
f    +1.99292207e- 1    +8.74742595e- 2    +8.22515778e- 1
x    +0.00000000e+ 0    +1.45912912e- 1    +1.97053995e- 1
f    +1.99292207e- 1    +1.02846461e- 1    +8.74742595e- 2
x    +1.45912912e- 1    +1.97053995e- 1    +2.53658473e- 1
f    +1.02846461e- 1    +8.74742595e- 2    +8.79586674e- 2
x    +1.97053995e- 1    +2.23864917e- 1    +2.53658473e- 1
f    +8.74742595e- 2    +8.47239289e- 2    +8.79586674e- 2
x    +2.23864917e- 1    +2.24209186e- 1    +2.53658473e- 1
f    +8.47239289e- 2    +8.47198050e- 2    +8.79586674e- 2
leave powell line search

exk=
+7.506090701e- 1      +5.484059900e- 1
function calls=  223      f=+8.471980505e- 2
enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +6.67618647e- 1    +8.47198050e- 2    +3.17122633e+ 1
x    -5.00000000e- 1    -2.40951709e- 1    +0.00000000e+ 0
```

```
f    +6.67618647e- 1    +7.35130638e- 1    +8.47198050e- 2
x    -2.40951709e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +7.35130638e- 1    +8.47198050e- 2    +3.17122633e+ 1
x    -2.40951709e- 1    -1.05313254e- 1    +0.00000000e+ 0
f    +7.35130638e- 1    +2.59605512e- 1    +8.47198050e- 2
x    -1.05313254e- 1    +0.00000000e+ 0    +5.57677252e- 2
f    +2.59605512e- 1    +8.47198050e- 2    +3.00737731e- 1
x    -1.05313254e- 1    -2.84889801e- 2    +0.00000000e+ 0
f    +2.59605512e- 1    +7.73461452e- 2    +8.47198050e- 2
x    -2.84889801e- 2    -1.94241130e- 2    +0.00000000e+ 0
f    +7.73461452e- 2    +7.37247439e- 2    +8.47198050e- 2
x    -2.84889801e- 2    -1.94241130e- 2    -1.80628616e- 2
f    +7.73461452e- 2    +7.37247439e- 2    +7.36331148e- 2
x    -1.94241130e- 2    -1.80628616e- 2    -1.76871512e- 2
f    +7.37247439e- 2    +7.36331148e- 2    +7.36294026e- 2
leave powell line search

enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +1.35159650e+ 0    +7.36294026e- 2    +3.66177946e+ 0
x    -5.00000000e- 1    -1.18687183e- 1    +0.00000000e+ 0
f    +1.35159650e+ 0    +1.26713259e- 1    +7.36294026e- 2
x    -1.18687183e- 1    -1.89046040e- 2    +0.00000000e+ 0
f    +1.26713259e- 1    +8.01006751e- 2    +7.36294026e- 2
x    -1.89046040e- 2    +0.00000000e+ 0    +1.53281810e- 1
f    +8.01006751e- 2    +7.36294026e- 2    +1.27601361e- 1
x    -1.89046040e- 2    +0.00000000e+ 0    +3.29869702e- 2
f    +8.01006751e- 2    +7.36294026e- 2    +6.60147583e- 2
x    +0.00000000e+ 0    +3.29869702e- 2    +7.02214288e- 2
f    +7.36294026e- 2    +6.60147583e- 2    +6.60870280e- 2
x    +3.29869702e- 2    +5.13114464e- 2    +7.02214288e- 2
f    +6.60147583e- 2    +6.46301536e- 2    +6.60870280e- 2
x    +5.13114464e- 2    +5.13673856e- 2    +7.02214288e- 2
f    +6.46301536e- 2    +6.46298230e- 2    +6.60870280e- 2
leave powell line search

enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +2.39683574e+ 0    +6.46298230e- 2    +2.25952851e- 1
x    +0.00000000e+ 0    +2.17651661e- 1    +5.00000000e- 1
f    +6.46298230e- 2    +1.36673381e- 2    +2.25952851e- 1
x    +0.00000000e+ 0    +1.68193504e- 1    +2.17651661e- 1
f    +6.46298230e- 2    +1.87468044e- 2    +1.36673381e- 2
x    +1.68193504e- 1    +2.17651661e- 1    +2.58630287e- 1
f    +1.87468044e- 2    +1.36673381e- 2    +1.30970854e- 2
x    +2.17651661e- 1    +2.45228229e- 1    +2.58630287e- 1
f    +1.36673381e- 2    +1.28173217e- 2    +1.30970854e- 2
x    +2.17651661e- 1    +2.43656158e- 1    +2.45228229e- 1
f    +1.36673381e- 2    +1.28169731e- 2    +1.28173217e- 2
x    +2.43656158e- 1    +2.44349228e- 1    +2.45228229e- 1
f    +1.28169731e- 2    +1.28163267e- 2    +1.28173217e- 2
leave powell line search

exk=
+9.140367955e- 1     +8.280966803e- 1
function calls=  254     f=+1.281632669e- 2
enter powell line search
```



```
x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +3.37269433e- 1    +1.28163267e- 2    +1.08940594e+ 1
x    -5.00000000e- 1    -2.35522849e- 1    +0.00000000e+ 0
f    +3.37269433e- 1    +2.94631474e- 1    +1.28163267e- 2
x    -2.35522849e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +2.94631474e- 1    +1.28163267e- 2    +1.08940594e+ 1
x    -2.35522849e- 1    -9.85948816e- 2    +0.00000000e+ 0
f    +2.94631474e- 1    +7.90189700e- 2    +1.28163267e- 2
x    -9.85948816e- 2    +0.00000000e+ 0    +3.82512049e- 2
f    +7.90189700e- 2    +1.28163267e- 2    +5.22225159e- 2
x    -2.22981982e- 2    +0.00000000e+ 0    +3.82512049e- 2
f    +1.05510169e- 2    +1.28163267e- 2    +5.22225159e- 2
x    -2.22981982e- 2    -1.44612306e- 2    +0.00000000e+ 0
f    +1.05510169e- 2    +9.85187024e- 3    +1.28163267e- 2
x    -2.22981982e- 2    -1.49989856e- 2    -1.44612306e- 2
f    +1.05510169e- 2    +9.85035126e- 3    +9.85187024e- 3
x    -1.49989856e- 2    -1.48421148e- 2    -1.44612306e- 2
f    +9.85035126e- 3    +9.85002192e- 3    +9.85187024e- 3
leave powell line search

enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +5.09365596e- 1    +9.85002192e- 3    +1.36174719e+ 0
x    -5.00000000e- 1    -1.15098807e- 1    +0.00000000e+ 0
f    +5.09365596e- 1    +2.69988521e- 2    +9.85002192e- 3
x    -1.15098807e- 1    -2.38172442e- 2    +0.00000000e+ 0
f    +2.69988521e- 2    +1.24036020e- 2    +9.85002192e- 3
x    -2.38172442e- 2    +0.00000000e+ 0    +1.05223873e- 1
f    +1.24036020e- 2    +9.85002192e- 3    +1.48895771e- 2
x    +0.00000000e+ 0    +3.26896878e- 2    +1.05223873e- 1
f    +9.85002192e- 3    +7.88204267e- 3    +1.48895771e- 2
x    +3.26896878e- 2    +3.65431495e- 2    +1.05223873e- 1
f    +7.88204267e- 3    +7.80377019e- 3    +1.48895771e- 2
x    +3.26896878e- 2    +3.65431495e- 2    +4.05821623e- 2
f    +7.88204267e- 3    +7.80377019e- 3    +7.76306883e- 3
x    +3.65431495e- 2    +4.05821623e- 2    +4.24478976e- 2
f    +7.80377019e- 3    +7.76306883e- 3    +7.75914115e- 3
x    +4.05821623e- 2    +4.22946913e- 2    +4.24478976e- 2
f    +7.76306883e- 3    +7.75910454e- 3    +7.75914115e- 3
leave powell line search

enter powell line search

x    -5.00000000e- 1    +0.00000000e+ 0    +5.00000000e- 1
f    +1.22472870e+ 0    +7.75910454e- 3    +1.06611481e- 1
x    +0.00000000e+ 0    +2.12437024e- 1    +5.00000000e- 1
f    +7.75910454e- 3    +1.12650226e- 3    +1.06611481e- 1
x    +0.00000000e+ 0    +1.25827769e- 1    +2.12437024e- 1
f    +7.75910454e- 3    +1.83637931e- 4    +1.12650226e- 3
x    +1.25827769e- 1    +1.52866887e- 1    +2.12437024e- 1
f    +1.83637931e- 4    +1.55626729e- 4    +1.12650226e- 3
x    +1.25827769e- 1    +1.41935393e- 1    +1.52866887e- 1
f    +1.83637931e- 4    +1.31514289e- 4    +1.55626729e- 4
x    +1.25827769e- 1    +1.41921066e- 1    +1.41935393e- 1
f    +1.83637931e- 4    +1.31511639e- 4    +1.31514289e- 4
leave powell line search

exk=
+9.981674552e- 1     +9.974703193e- 1
```

```
function calls=  285      f=+1.315116388e- 4
enter powell line search

x   -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +8.65262793e- 2   +1.31511639e- 4   +2.88888773e+ 0
x   -5.00000000e- 1   -2.35480607e- 1   +0.00000000e+ 0
f   +8.65262793e- 2   +6.52569691e- 2   +1.31511639e- 4
x   -2.35480607e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +6.52569691e- 2   +1.31511639e- 4   +2.88888773e+ 0
x   -2.35480607e- 1   -1.00941089e- 1   +0.00000000e+ 0
f   +6.52569691e- 2   +2.46044393e- 2   +1.31511639e- 4
x   -1.00941089e- 1   +0.00000000e+ 0   +4.27581329e- 1
f   +2.46044393e- 2   +1.31511639e- 4   +1.81653561e+ 0
x   -1.00941089e- 1   -3.62028768e- 2   +0.00000000e+ 0
f   +2.46044393e- 2   +5.03345806e- 3   +1.31511639e- 4
x   -3.62028768e- 2   +0.00000000e+ 0   +2.28423243e- 2
f   +5.03345806e- 3   +1.31511639e- 4   +9.44340980e- 4
x   +0.00000000e+ 0   +5.27715206e- 3   +2.28423243e- 2
f   +1.31511639e- 4   +7.51596616e- 6   +9.44340980e- 4
x   +5.27715206e- 3   +6.13143534e- 3   +2.28423243e- 2
f   +7.51596616e- 6   +4.03085960e- 6   +9.44340980e- 4
x   +6.13143534e- 3   +6.29799417e- 3   +2.28423243e- 2
f   +4.03085960e- 6   +3.90241172e- 6   +9.44340980e- 4
leave powell line search

enter powell line search

x   -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +4.70083609e- 1   +3.90241172e- 6   +4.13609996e- 1
x   +0.00000000e+ 0   +1.59767228e- 2   +5.00000000e- 1
f   +3.90241172e- 6   +9.63963557e- 5   +4.13609996e- 1
x   +0.00000000e+ 0   +6.28269126e- 3   +1.59767228e- 2
f   +3.90241172e- 6   +2.54916868e- 5   +9.63963557e- 5
x   -3.93726176e- 3   +0.00000000e+ 0   +6.28269126e- 3
f   +2.37243130e- 7   +3.90241172e- 6   +2.54916868e- 5
x   -3.93726176e- 3   -3.86725285e- 3   +0.00000000e+ 0
f   +2.37243130e- 7   +2.35405960e- 7   +3.90241172e- 6
leave powell line search

enter powell line search

x   -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +9.91243824e- 1   +2.35405960e- 7   +1.26989499e+ 1
x   -5.00000000e- 1   -2.13797312e- 1   +0.00000000e+ 0
f   +9.91243824e- 1   +5.38525552e- 1   +2.35405960e- 7
x   -2.13797312e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +5.38525552e- 1   +2.35405960e- 7   +1.26989499e+ 1
x   -2.13797312e- 1   -7.46965846e- 2   +0.00000000e+ 0
f   +5.38525552e- 1   +9.61686065e- 2   +2.35405960e- 7
x   -7.46965846e- 2   +0.00000000e+ 0   +3.53676677e- 2
f   +9.61686065e- 2   +2.35405960e- 7   +2.76322991e- 2
x   -3.09966182e- 3   +0.00000000e+ 0   +3.53676677e- 2
f   +2.09657945e- 4   +2.35405960e- 7   +2.76322991e- 2
x   -3.09966182e- 3   -1.89423226e- 5   +0.00000000e+ 0
f   +2.09657945e- 4   +3.26124081e- 7   +2.35405960e- 7
leave powell line search

exk=
+9.999265447e- 1      +9.999010563e- 1
function calls=  314      f=+2.354059596e- 7
```



```
enter powell line search

x   -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +4.83619973e- 1   +2.35405960e- 7   +4.01694041e- 1
x   +0.00000000e+ 0   +2.31347224e- 2   +5.00000000e- 1
f   +2.35405960e- 7   +1.30852786e- 4   +4.01694041e- 1
x   +0.00000000e+ 0   +9.87987546e- 3   +2.31347224e- 2
f   +2.35405960e- 7   +2.41686457e- 5   +1.30852786e- 4
x   -4.04605671e- 5   +0.00000000e+ 0   +9.87987546e- 3
f   +2.36125183e- 7   +2.35405960e- 7   +2.41686457e- 5
leave powell line search

enter powell line search

x   -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +9.91243824e- 1   +2.35405960e- 7   +1.26989499e+ 1
x   -5.00000000e- 1   -2.13797312e- 1   +0.00000000e+ 0
f   +9.91243824e- 1   +5.38525552e- 1   +2.35405960e- 7
x   -2.13797312e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +5.38525552e- 1   +2.35405960e- 7   +1.26989499e+ 1
x   -2.13797312e- 1   -7.46965846e- 2   +0.00000000e+ 0
f   +5.38525552e- 1   +9.61686065e- 2   +2.35405960e- 7
x   -7.46965846e- 2   +0.00000000e+ 0   +3.53676677e- 2
f   +9.61686065e- 2   +2.35405960e- 7   +2.76322991e- 2
x   -3.09966182e- 3   +0.00000000e+ 0   +3.53676677e- 2
f   +2.09657945e- 4   +2.35405960e- 7   +2.76322991e- 2
x   -3.09966182e- 3   -1.89423226e- 5   +0.00000000e+ 0
f   +2.09657945e- 4   +3.26124081e- 7   +2.35405960e- 7
leave powell line search

exk=
+9.999265447e- 1      +9.999010563e- 1
function calls=  330      f=+2.354059596e- 7
ilp=0

enter powell line search

x   -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +7.94775955e- 1   +1.74202396e- 1   +1.51774806e+ 0
x   -5.00000000e- 1   -9.20224311e- 2   +0.00000000e+ 0
f   +7.94775955e- 1   +1.64786700e- 1   +1.74202396e- 1
x   -9.20224311e- 2   -6.15471774e- 2   +0.00000000e+ 0
f   +1.64786700e- 1   +1.63657073e- 1   +1.74202396e- 1
x   -9.20224311e- 2   -6.86011873e- 2   -6.15471774e- 2
f   +1.64786700e- 1   +1.63547570e- 1   +1.63657073e- 1
x   -6.86011873e- 2   -6.85308622e- 2   -6.15471774e- 2
f   +1.63547570e- 1   +1.63547499e- 1   +1.63657073e- 1
leave powell line search

enter powell line search

x   -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f   +4.46816802e- 1   +1.63547499e- 1   +1.60802574e+ 1
x   -5.00000000e- 1   -2.41257107e- 1   +0.00000000e+ 0
f   +4.46816802e- 1   +1.81085320e- 1   +1.63547499e- 1
x   -2.41257107e- 1   -1.01585212e- 1   +0.00000000e+ 0
f   +1.81085320e- 1   +4.59589949e- 3   +1.63547499e- 1
x   -1.17528150e- 1   -1.01585212e- 1   +0.00000000e+ 0
f   +9.75931331e- 3   +4.59589949e- 3   +1.63547499e- 1
x   -1.17528150e- 1   -1.01585212e- 1   -9.94793624e- 2
```

```
f    +9.75931331e- 3   +4.59589949e- 3   +4.37712990e- 3
x    -1.01585212e- 1   -9.94793624e- 2   -9.62705072e- 2
f    +4.59589949e- 3   +4.37712990e- 3   +4.26481874e- 3
x    -9.94793624e- 2   -9.65247629e- 2   -9.62705072e- 2
f    +4.37712990e- 3   +4.26383701e- 3   +4.26481874e- 3
leave powell line search

enter powell line search

x    -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f    +3.65568441e+ 0   +4.26383701e- 3   +1.55903568e- 1
x    +0.00000000e+ 0   +2.30063459e- 1   +5.00000000e- 1
f    +4.26383701e- 3   +7.87594598e- 2   +1.55903568e- 1
x    -5.00000000e- 1   +0.00000000e+ 0   +2.30063459e- 1
f    +3.65568441e+ 0   +4.26383701e- 3   +7.87594598e- 2
x    +0.00000000e+ 0   +9.95335700e- 2   +2.30063459e- 1
f    +4.26383701e- 3   +3.16997436e- 2   +7.87594598e- 2
x    -3.23779125e- 1   +0.00000000e+ 0   +9.95335700e- 2
f    +1.03156526e+ 0   +4.26383701e- 3   +3.16997436e- 2
x    +0.00000000e+ 0   +3.28486725e- 2   +9.95335700e- 2
f    +4.26383701e- 3   +8.86217493e- 3   +3.16997436e- 2
x    -1.79814217e- 2   +0.00000000e+ 0   +3.28486725e- 2
f    +4.93130425e- 3   +4.26383701e- 3   +8.86217493e- 3
x    -1.79814217e- 2   -3.66392516e- 3   +0.00000000e+ 0
f    +4.93130425e- 3   +4.19032871e- 3   +4.26383701e- 3
x    -1.79814217e- 2   -4.34363802e- 3   -3.66392516e- 3
f    +4.93130425e- 3   +4.18789859e- 3   +4.19032871e- 3
x    -1.79814217e- 2   -4.44440963e- 3   -4.34363802e- 3
f    +4.93130425e- 3   +4.18783945e- 3   +4.18789859e- 3
leave powell line search

exk=
+9.388300776e- 1      +8.792898506e- 1
function calls=  359      f=+4.187839455e- 3
enter powell line search

x    -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f    +1.04568392e+ 0   +4.18783945e- 3   +1.25797601e- 1
x    +0.00000000e+ 0   +1.97721971e- 1   +5.00000000e- 1
f    +4.18783945e- 3   +1.61424016e- 3   +1.25797601e- 1
x    +0.00000000e+ 0   +1.06538539e- 1   +1.97721971e- 1
f    +4.18783945e- 3   +5.79419873e- 4   +1.61424016e- 3
x    +1.06538539e- 1   +1.27318423e- 1   +1.97721971e- 1
f    +5.79419873e- 4   +6.10214069e- 4   +1.61424016e- 3
x    +1.06538539e- 1   +1.11541454e- 1   +1.27318423e- 1
f    +5.79419873e- 4   +5.75656384e- 4   +6.10214069e- 4
x    +1.11641454e- 1   +1.11694741e- 1   +1.27318423e- 1
f    +5.75656384e- 4   +5.75656602e- 4   +6.10214069e- 4
leave powell line search

enter powell line search

x    -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f    +5.04930240e+ 0   +5.75656384e- 4   +1.87534388e- 1
x    +0.00000000e+ 0   +2.32145727e- 1   +5.00000000e- 1
f    +5.75656384e- 4   +1.78365378e- 1   +1.87534388e- 1
x    -5.00000000e- 1   +0.00000000e+ 0   +2.32145727e- 1
f    +5.04930240e+ 0   +5.75656384e- 4   +1.78365378e- 1
x    +0.00000000e+ 0   +9.02650366e- 2   +2.32145727e- 1
f    +5.75656384e- 4   +5.15296166e- 2   +1.78365378e- 1
```



```
x    -1.53740527e- 1   +0.00000000e+ 0   +9.02650366e- 2
f    +2.17425875e- 1   +5.75656384e- 4   +5.15296166e- 2
x    +0.00000000e+ 0   +1.02615716e- 2   +9.02650366e- 2
f    +5.75656384e- 4   +2.52122263e- 3   +5.15296166e- 2
x    -1.50994412e- 2   +0.00000000e+ 0   +1.02615716e- 2
f    +2.57923024e- 4   +5.75656384e- 4   +2.52122263e- 3
x    -1.50994412e- 2   -9.13278281e- 3   +0.00000000e+ 0
f    +2.57923024e- 4   +1.32243305e- 9   +5.75656384e- 4
x    -9.13278281e- 3   -9.04481136e- 3   +0.00000000e+ 0
f    +1.32243305e- 9   +4.54707538e- 8   +5.75656384e- 4
leave powell line search

exk=
+1.000032827e+ 0      +1.000064090e+ 0
function calls=  379      f=+1.322433052e- 9
enter powell line search

x    -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f    +4.83524688e- 1   +1.32243305e- 9   +4.02941100e- 1
x    +0.00000000e+ 0   +2.27260850e- 2   +5.00000000e- 1
f    +1.32243305e- 9   +1.28208731e- 4   +4.02941100e- 1
x    +0.00000000e+ 0   +9.68073448e- 3   +2.27260850e- 2
f    +1.32243305e- 9   +2.35880098e- 5   +1.28208731e- 4
x    -1.18256944e- 4   +0.00000000e+ 0   +9.68073448e- 3
f    +1.21880750e- 9   +1.32243305e- 9   +2.35880098e- 5
leave powell line search

enter powell line search

x    -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f    +5.31960803e+ 0   +1.21880750e- 9   +1.91352373e- 1
x    +0.00000000e+ 0   +2.32638927e- 1   +5.00000000e- 1
f    +1.21880750e- 9   +1.71812737e- 1   +1.91352373e- 1
x    -5.00000000e- 1   +0.00000000e+ 0   +2.32638927e- 1
f    +5.31960803e+ 0   +1.21880750e- 9   +1.71812737e- 1
x    +0.00000000e+ 0   +9.25414041e- 2   +2.32638927e- 1
f    +1.21880750e- 9   +4.56684735e- 2   +1.71812737e- 1
x    -9.47989002e- 2   +0.00000000e+ 0   +9.25414041e- 2
f    +8.25353647e- 2   +1.21880750e- 9   +4.56684735e- 2
x    +0.00000000e+ 0   +1.23842341e- 2   +9.25414041e- 2
f    +1.21880750e- 9   +1.04884520e- 3   +4.56684735e- 2
x    -2.11103048e- 3   +0.00000000e+ 0   +1.23842341e- 2
f    +3.20099889e- 5   +1.21880750e- 9   +1.04884520e- 3
x    +0.00000000e+ 0   +4.50175580e- 5   +1.23842341e- 2
f    +1.21880750e- 9   +1.16164279e- 8   +1.04884520e- 3
leave powell line search

exk=
+9.999740869e- 1      +9.999458343e- 1
function calls=  396      f=+1.218807499e- 9
flp=1
```

```
enter powell line search

x    -5.00000000e- 1   +0.00000000e+ 0   +5.00000000e- 1
f    +8.03669536e+ 0   +2.35405960e- 7   +6.09261265e+ 1
x    -5.00000000e- 1   -1.91731684e- 1   +0.00000000e+ 0
f    +8.03669536e+ 0   +2.79697224e+ 0   +2.35405960e- 7
x    -1.91731684e- 1   +0.00000000e+ 0   +1.41782658e+ 0
f    +2.79697224e+ 0   +2.35405960e- 7   +1.23418637e+ 3
x    -1.91731684e- 1   -8.26012017e- 2   +0.00000000e+ 0
f    +2.79697224e+ 0   +6.56738974e- 1   +2.35405960e- 7
x    -8.26012017e- 2   +0.00000000e+ 0   +2.40629022e- 2
f    +6.56738974e- 1   +2.35405960e- 7   +6.81209192e- 2
x    -1.97195585e- 3   +0.00000000e+ 0   +2.40629022e- 2
f    +4.58036338e- 4   +2.35405960e- 7   +6.81209192e- 2
x    -1.97195585e- 3   +0.00000000e+ 0   +6.28247115e- 7
f    +4.58036338e- 4   +2.35405960e- 7   +2.28997989e- 7
leave powell line search
```


(vii) The next method illustrates the use of non-accurate line searches and is an implementation of Dixon[13]. To vary the theme to more recent research a rational line search is used.

```
PROGRAM dixon
BEGIN
   [1:2] REAL x;

   PROC f1=(REF[ ]REAL x)REAL:
   BEGIN
     REAL w:=x[1]*x[1]-x[2];
     100.0*w*w+(1-x[1])*(1-x[1])
   END;

   PROC g1=(INT i,REF[ ]REAL x)REAL:
   BEGIN
      CASE i IN
         400.0*x[1]*(x[1]*x[1]-x[2])-2.0*(1.0-x[1]),
         -200.0*(x[1]*x[1]-x[2])
      ESAC
   END;

   PROC amax1=(REAL p1,p2,p3,p4,p5)REAL:
   BEGIN
     REAL am:=p1;
     (p2>am!am:=p2);
     (p3>am!am:=p3);
     (p4>am!am:=p4);
     (p5>am!am:=p5);
     am
   END;

   PROC order=(REF[ ]REAL x,p) VOID :
   BEGIN
      INT n:=UPB x,j;
      REAL min,w;
      FOR i1 TO n-1 DO
         j:=i1;min:=x[j];
         FOR i2 FROM j+1 TO n DO
            (x[i2]<min!min:=x[i2];j:=i2)
         OD;
         (j/=i1!w:=x[j];x[j]:=x[i1];x[i1]:=w;
               w:=p[j];p[j]:=p[i1];p[i1]:=w)
      OD
   END;

   PROC appline=(PROC(REF[ ]REAL)REAL f,REF[ ]REAL pp,x0,
               REAL h0,er,REF REAL fv)REAL:
     BEGIN
        REAL p1,pp1,p2,pp2,p3,pp3,p4,pp4,p5,pp5,p6,pp6,p7,pp7,amin,max,
              amin,ggg,t,a1,b1,c1,d1,e1,nh,q,s,miue,lnda,sqt,nq,mns,
              sr,qr,qq,mu;
        REAL scale;
        REAL h:=h0;
        INT n:=UPB x0,iterat:=0,j,iz,maxit,qqcnt;
        [1:n]REAL xar;
        BOOL initon:=TRUE,con:=FALSE,f0set:=TRUE,minfail:=FAUSE;
        BOOL sing,bflat,qqon,outset;
        outset:=TRUE;
        maxit:=1;
        REAL f0;
        [1:n]REAL d;
        d:=pp;
        [1:5,1:5] REAL a;
        [1:5] REAL b,y,x,p;
        FORMAT ft1=$l"Initial amin="+d.7de+zd7x,
                     "at point ggg="+d.7de+zd$,
               ft2=$l"x["zd,"]="+d.7de+zd7x,
                     "p["zd,"]="+d.7de+zd$,
               ft3=$l"its="3zd5x,"min="+d.7de+zd7x,"qq="+d.7de+zd$,
               ft5=$l"a1="+d.4de+zd3x,"b1="+d.4de+zd3x,"c1="+d.4de+zdl,
                   "d1="+d.4de+zd3x,"e1="+d.4de+zd$,
               ft6=$l"a1="+d.4de+zd3x,"b1="+d.4de+zd3x,"c1="+d.4de+zd3x,
                   "e1="+d.4de+zd$,
               ft7=$l"a1="+d.4de+zd3x,"b1="+d.4de+zd3x,"c1="+d.4de+zd$,
               ft4=$l"amin="+d.7de+zd7x,"qq="+d.7de+zd";
        max:= ABS pp[1];
        FOR i1 FROM 2 TO n DO (max< ABS pp[i1]!max:=ABS pp[i1]) OD;
        FOR i1 TO n DO d[i1]:=pp[i1]/max OD;
        scale:=max;

        PROC tr=(REAL ex) VOID:
        BEGIN
          FOR i1 TO n DO xar[i1]:=x0[i1]+ex*d[i1] OD
        END;

        PROC init=VOID:
        BEGIN
           initon:=FALSE;
           tr(p1);pp1:=f(xar);
           (f0set!f0:=pp1;f0set:=FALSE);
           p2:=p1+2.0*h;
           tr(p2);pp2:=f(xar);
           IF pp1<pp2 THEN
              p3:=p1-2.0*h;
              tr(p3);pp3:=f(xar);
              IF pp3<pp1 THEN
                 p4:=p3-4.0*h;
                 tr(p4);pp4:=f(xar);
                 IF pp4<pp3 THEN
                    p5:=p4-8.0*h;
                    tr(p5);pp5:=f(xar);
                    IF pp5<pp4 THEN
                       p7:=(p4+p5)/2.0;
                       tr(p7);pp7:=f(xar);
                       IF pp5>=pp7 THEN
                          p6:=(p1+p2)/2.0;
                          tr(p5);pp6:=f(xar);
                          IF pp6>=pp1 THEN
                             initon:=TRUE;
                             p1 MINUSAB 25.0*h
                          FI
                       FI
                    FI
                 ELSE
                    p5:=p4+2.0*h;
                    tr(p5);pp5:=f(xar)
                 FI
```

```
          ELSE
             p4:=p3+h;
             tr(p4);pp4:=f(xar);
             IF pp4<pp1 THEN
                p5:=p1-h/2.0;
                tr(p5);pp5:=f(xar)
             ELSE
                p5:=p1+h;
                tr(p5);pp5:=f(xar)
             FI
          FI
       ELSE
          p3:=p2+2.0*h;
          tr(p3);pp3:=f(xar);
          IF pp2<=pp3 THEN
             p4:=p3-h;
             tr(p4);pp4:=f(xar);
             IF pp4<pp2 THEN
                p5:=p4+h/2.0;
                tr(p5);pp5:=f(xar)
             ELSE
                p5:=p2-h;
                tr(p5);pp5:=f(xar)
             FI
          ELSE
             p4:=p3+4.0*h;
             tr(p4);pp4:=f(xar);
             IF pp4<pp3 THEN
                p5:=p4+8.0*h;
                tr(p5);pp5:=f(xar);
                IF pp5<pp4 THEN
                   p7:=(p4+p5)/2.0;
                   tr(p7);pp7:=f(xar);
                   IF pp7>=pp5 THEN
                      p6:=(p1+p2)/2.0;
                      tr(p6);pp6:=f(xar);
                      IF pp6>=pp1 THEN
                         initon:=TRUE;
                         p1 PLUSAB 25.0*h
                      FI
                   FI
                FI
             ELSE
                p5:=p4-2.0*h;
                tr(p5);pp5:=f(xar)
             FI
          FI
    FI
 END;

 PROC gauss=(REF[,]REAL a,REF[ ]REAL y,b,REF BOOL sing) VOID:
 BEGIN
    INT n:=UPB y,l;
    REAL am,w;
    sing:=FALSE;
    clear(y);
    FOR i1 TO n-1 WHILE NOT sing DO
       am:=ABS a[i1,i1];l:=i1;
       FOR i2 FROM i1+1 TO n DO
          (am<ABS a[i2,i1]!am:=ABS a[i2,i1];l:=i2)
```



```
       OD;
       IF NOT (sing:=am<1.0e-8) THEN
          IF l/=i1 THEN
             FOR i2 TO n DO(w:=a[l,i2];a[l,i2]:=a[i1,i2];a[i1,i2]
             OD;
             w:=b[l];b[l]:=b[i1];b[i1]:=w
          FI;
          FOR i2 FROM i1+1 TO n DO
             am:=a[i2,i1]/a[i1,i1];
             FOR i3 FROM i1+1 TO n DO a[i2,i3] MINUSAB am*a[i1,i3
             b[i2] MINUSAB am*b[i1]
          OD
       FI
    OD;
    (NOT sing!(ABS a[n,n]<1.0e-8!sing:=TRUE));
    IF NOT sing THEN
       y[n]:=b[n]/a[n,n];
       FOR i2 FROM n-1 BY -1 TO 1 DO
          y[i2]:=b[i2];
          FOR i3 FROM i2+1 TO n DO y[i2] MINUSAB y[i3]*a[i2,i3] O
          y[i2] DIVAB a[i2,i2]
       OD
    FI
 END;

 PROC root =(REAL a,b,c,REF REAL d,e)VOID:
 BEGIN
   REAL srt,x1,x2;
   REAL sc,aa,bb,cc;
   sc:=ABS a;
   (ABS b>sc!sc:=ABS b);
   (ABS c >sc!sc:=ABS c);
    aa:=a/sc;bb:=b/sc;cc:=c/sc;
    srt:=sqrt(aa*aa-4.0*bb*cc);
    x1:=(-aa-srt)/2.0/bb;
    x2:=(-aa+srt)/2.0/bb;
    (x1>x2!d:=x2;e:=x1!d:=x1;e:=x2)
END;

 PROC rej=(REF BOOL sing,INT n)VOID:
 BEGIN
   REAL max:=ABS y[1],min;
   min:=max;
   FOR i1 FROM 2 TO n DO
      IF ABS y[i1]>max THEN max:=ABS y[i1] FI;
      IF ABS y[i1]<min THEN min:=ABS y[i1] FI
   OD;
   IF NOT sing THEN
    (min/max<1.0e-8 OR max>1.0e+8!sing:=TRUE)
   FI
 END;

 PROC flat=BOOL:
 BEGIN
    REAL mx:=p[1],mn:=p[1];
    INT jx:=1,jn:=1;
    FOR i1 FROM 2 TO 5 DO
       (p[i1]>mx!mx:=p[i1];jx:=i1);
       (p[i1]<mn!mn:=p[i1];jn:=i1)
    OD;
```

```
      ABS((mx-mn)/(x[jx]-x[jn]))<0.1*ABS mx
   END;

   p1:=0.0;
   WHILE initon DO init OD;
   x:=(p1,p2,p3,p4,p5);
   p:=(pp1,pp2,pp3,pp4,pp5);
   amin:=p[1];gog:=x[1];
   FOR i FROM 2 TO 5 DO
      IF p[i]<amin THEN amin:=p[i];gog:=x[i] FI
   OD;
   IF outset THEN
      print((newline,"Enter Moufft"));
      print((newline,"d=",d,newline,"x0=",x0));
      printf((ft1,amin,gog))
   FI;
   qqcnt:=0;
   qqon:=FALSE;
   TO maxit WHILE NOT con DO
      order(x,p);
      IF outset THEN
         FOR i1 TO 5 DO printf((ft2,i1,x[i1],i1,p[i1])) OD
      FI;
      bflat:=flat;
      IF bflat THEN
         sing:=TRUE;
         (outset!print((newline,"flat")))
      ELIF qqon THEN
         sing:=TRUE;
         (outset!print((newline,"qqon")))
      ELSE
         FOR i TO 5 DO
            t:=x[i];a[i,1]:=t*t;a[i,2]:=t+t;a[i,3]:=1.0;
            a[i,4]:=-p[i]*t*2.0;
            a[i,5]:=-p[i];b[i]:=p[i]*t*t
         OD;
         gauss(a,y,b,sing);
         a1:=y[1];b1:=y[2];c1:=y[3];d1:=y[4];e1:=y[5];
         (outset!printf((ft5,a1,b1,c1,d1,e1)));
         rej(sing,5)
      FI;
      IF sing THEN
         iz:=IF p[2]<p[1] AND (p[3]<p[4] OR p[2]<p[3])
             THEN 0
             ELIF p[4]<p[5] AND (p[3]<p[2] OR p[4]<p[3])
             THEN 1 ELIF p[1]<p[5] THEN 0 ELSE 1
             FI;
         IF bflat OR qqon THEN sing:=TRUE
         ELSE
            FOR i TO 4 DO
               t:=x[iz+i];a[i,1]:=t*t;a[i,2]:=t+t;a[i,3]:=1.0;
               a[i,4]:=-p[iz+i];b[i]:=p[iz+i]*t
            OD;
            gauss(a[1:4,1:4],y[1:4],b[1:4],sing);
            a1:=y[1];b1:=y[2];c1:=y[3];e1:=y[4];
            (outset!printf((ft6,a1,b1,c1,e1)));
            rej(sing,4)
         FI;
         IF sing THEN
            IF p[iz+3]<p[iz+2] AND p[iz+3]<p[iz+4] THEN iz PLUSAB 1
```



```
            ELIF p[iz+2]<p[iz+1] AND p[iz+2]<p[iz+3] THEN SKIP
            ELIF p[iz+1]>p[iz+4] THEN iz PLUSAB 1
            FI;
            FOR i TO 3 DO
               t:=x[iz+i];a[i,1]:=t*t;a[i,2]:=t+t;
               a[i,3]:=1.0;b[i]:=p[iz+i]
            OD;
            gauss(a[1:3,1:3],y[1:3],b[1:3],sing);
            a1:=y[1];b1:=y[2];c1:=y[3];
            IF outset THEN
               print((newline,"Quadratic Case",newline));
               printf((ft7,a1,b1,c1))
            FI;
            IF a1>0.0 THEN min:=-b1/a1 ELSE minfail:=TRUE FI
         ELSE
            mu:=c1-2*b1*e1+a1*e1*e1;
            (outset!print((newline,"Special Rational",newline)));
            IF mu>0.0 AND a1>0.0 THEN
               min:=-e1+sqrt(mu/a1)
            ELIF mu<0.0 AND a1<0.0 THEN
               min:=-e1-sqrt(mu/a1)
            ELSE minfail:=TRUE
            FI
         FI
      ELSE
         hh:=a1*d1-b1;g:=a1*e1-c1;s:=e1*b1-c1*d1;
         IF d1*d1>=e1 THEN
            miue:=c1-a1*e1;
            lnda:=b1-a1*d1;
            sqt:=sqrt(d1*d1-e1);
            r:=miue-2.0*lnda*(d1-sqt);
            q:=miue-2.0*lnda*(d1+sqt);
            IF ABS lnda<1.0e-35 THEN
               min:=-d1
            ELIF (r>0.0 AND q<0.0) OR (r<0.0 AND q>0.0) THEN
               min:=-miue/2.0/lnda
            ELSE
               min:=(-sqrt(r*q)-miue)/2.0/lnda
            FI
         ELIF a1<0.0 THEN
            IF b1/a1-d1<0.0 THEN
               root(g,hh,s,sr,qr);min:=sr
            ELIF b1/a1-d1>0.0 THEN
               root(g,hh,s,sr,qr);min:=qr
            ELSE min:=-d1
            FI
         ELIF a1>0.0 THEN
            IF b1/a1-d1<0.0 THEN
               root(g,hh,s,sr,qr);min:=qr
            ELIF b1/a1-d1>0.0 THEN
               root(g,hh,s,sr,qr);min:=sr
            ELSE min:=-d1
            FI
         ELSE
            IF b1<0.0 THEN
               root(g,hh,s,sr,qr);min:=-qr
            ELIF b1>0.0 THEN
               root(g,hh,s,sr,qr);min:=sr
            ELSE min:=-d1
            FI
```

```
            FI
         FI;
         tr(min);qq:=f(xar);
         iterat PLUSAB 1;
         ans:=min;fv:=qq;
         (outset!printf((ft3,iterat,min,qq)));
         IF qq>amax1(pp1,pp2,pp3,pp4,pp5) OR minfail THEN
            (outset!print((newline,"qq>amax1     minfail=",minfail)));
            h DIVAB 2.0;
            ((qqcnt PLUSAB 1)>=2!qqon:=TRUE;qqcnt:=0);
            iterat:=0;
            initon:=TRUE;
            WHILE initon DO init OD;
            x:=(p1,p2,p3,p4,p5);
            p:=(pp1,pp2,pp3,pp4,pp5);
            amin:=p[1];gqg:=x[1];
            FOR i FROM 2 TO 5 DO
               IF p[i]<amin THEN amin:=p[i];gqg:=x[i] FI
            OD;
            (outset!printf((ft1,amin,gqg)));
            con:=FALSE
         ELSE
            (outset!printf((ft4,amin,qq)));
            IF ABS (qq-amin)<=er*er*(ABS amin+0.5)*10 THEN
               con:=TRUE
            ELSE
               IF qq<=amin THEN amin:=qq;gqg:=min FI;
               j:=1;max:=p[1];
               FOR i FROM 2 TO 5 DO (p[i]>max!max:=p[i];j:=i) OD ;
               x[j]:=min;
               p[j]:=qq
            FI
         FI
      OD;
      (outset!print((newline,"Leave Mouffti",newline)));
      ans/scale
   END;

PROC dixon=(PROC(REF[ ]REAL)REAL f,
            PROC(INT,REF[ ]REAL)REAL g,
            INT n,REAL err,REF[ ]REAL x0)REAL:
BEGIN
   [1:n] REAL x,w,ge,d,p,x1,ge1,y,hy,yh,w1;
   [1:n,1:n] REAL h,n1;
   REAL s,ef,lamda,ef1,ep1,s1,s2,hh,er1;
   BOOL con:=TRUE,carryon:=TRUE,hset:=FALSE;
   INT maxits;
   ep1:=10.0*err-12;
   er1:=1.0e-2;
   hh:=0.25;
   maxits:=100;
   FORMAT ft1=$l5(+d.7de+zd5x)$;

   PROC dumpout=(REF[ ]REAL x,ge,REAL ef,REF[ ]REAL p)VOID:
   BEGIN
      print((newline,"x="));printf((ft1,x));
      print((newline,"ge="));printf((ft1,ge));
      print((newline,"ef=",ef));
      print((newline,"p="));printf((ft1,p))
   END;
```



```

   PROC uph=VOID:
   BEGIN
      IF
         s:=0.0;
         FOR i1 TO n DO s PLUSAB d[i1]*y[i1] OD;
         s>ep1
      THEN
         FOR i1 TO n DO
            s1:=0.0;
            FOR i2 TO n DO s1 PLUSAB h[i1,i2]*y[i2] OD;
            hy[i1]:=s1
         OD;
         FOR i1 TO n DO
            s1:=0.0;
            FOR i2 TO n DO s1 PLUSAB y[i2]*h[i2,i1] OD;
            yh[i1]:=s1
         OD;
         s1:=0.0;
         FOR i1 TO n DO s1 PLUSAB y[i1]*hy[i1] OD;
         s1:=s1/s+1.0;
         FOR i1 TO n DO
            FOR i2 TO n DO
               h1[i1,i2]:=h[i1,i2]-
                  (d[i1]*yh[i2]+hy[i1]*d[i2]-s1*d[i1]*d[i2])/s
            OD
         OD
      ELSE
         print((newline,"s<ep1 condition set",newline))
      FI
   END;

   PROC uper=VOID:
   BEGIN
      IF ABS(ef-ef1)<er1*(ABS ef1 + 0.5) THEN
         IF hset THEN
            hh DIVAB 2.0;hset:=FALSE;er1 DIVAB 10.0
         ELSE hset:=TRUE
         FI
      FI
   END;

   x:=x0;
   clear(h);
   FOR i1 TO n DO h[i1,i1]:=1.0 OD;
   clear(w);
   ef1:=f(x);
   FOR i1 TO n DO ge[i1]:=g(i1,x0) OD;
   TO maxits WHILE carryon DO
      TO n WHILE con DO
         FOR i1 TO n DO
            s:=0.0;
            FOR i2 TO n DO s MINUSAB h[i1,i2]*ge[i2] OD;
            p[i1]:=s+w[i1]
         OD;
         ef:=ef1;
         s:=0.0;
         FOR i1 TO n DO s PLUSAB p[i1]*ge[i1] OD;
         IF -s<err THEN
            print((newline,"4.2 condition set"));
```

```
                dumpout(x,ge,ef,p)
            FI;
            lamda:=appline(f,p,x,hh,err,ef1);
            FOR i1 TO n DO x1[i1]:=x[i1]+lamda*p[i1] OD;
            FOR i1 TO n DO ge1[i1]:=g(i1,x1) OD;
            FOR i1 TO n DO d[i1]:=x1[i1]-x[i1] OD;
            FOR i1 TO n DO y[i1]:=ge1[i1]-ge[i1] OD;
            uph;
            print((newline,"h="));
            FOR i1 TO n DO printf((ft1,h1[i1, ])) OD;
            dumpout(x1,ge1,ef1,p);
            s1:=0.0;
            FOR i1 TO n DO s1 PLUSAB d[i1]*(ef1<ef!ge1[i1]!ge[i1]) OD;
            FOR i1 TO n DO w1[i1] :=w[i1]+d[i1]*s1/s OD;
            w:=w1;h:=h1;
            uper;
            (ef1<ef!x:=x1;ge:=ge1);
            TO 5 DO newline(stand out) OD
        OD;
    l2:clear(w);
    l1:FOR i1 TO n DO
            s:=0.0;
            FOR i2 TO n DO s MINUSAB h[i1,i2]*ge[i2] OD;
            p[i1]:=s
        OD;
        ef:=ef1;
        s:=0.0;
        FOR i1 TO n DO  s PLUSAB p[i1]*ge[i1] OD;
        IF -s<err THEN
           print((newline,"Tran 7 set",newline));
           clear(h);
           FOR i1 TO n DO h[i1,i1]:=1.0  OD;
           GOTO l1
        FI;
        lamda:=appline(f,p,x,hh,err,ef1);
        FOR i1 TO n DO x1[i1]:=x[i1]+lamda*p[i1] OD;
        FOR i1 TO n DO ge1[i1]:=g(i1,x1) OD;
        FOR i1 TO n DO d[i1]:=x1[i1]-x[i1] OD;
        FOR i1 TO n DO y[i1]:=ge1[i1]-ge[i1] OD;
        print((newline,"ge1="));
        printf((ft1,ge1));
        print((newline,"d="));
        printf((ft1,d));
        print((newline,"y="));
        printf((ft1,y));
        uph;
        print((newline,"End of major step  h="));
        FOR i1 TO n DO printf((ft1,h1[i1, ])) OD;
        dumpout(x1,ge1,ef1,p);
        uper;
```



```
            s1:=0.0;
            FOR i1 TO n DO s1 PLUSAB ge1[i1]*ge1[i1] OD;
            s2:=0.0;
            FOR i1 TO n DO s2 PLUSAB d[i1]*d[i1] OD;
            carryon:=(s1>err) AND (s2>err*err)
                     AND  (ABS(ef-ef1)>err*err*(ABS ef+ 0.5));
            x:=x1;ge:=ge1;h:=h1;
            TO 10 DO newline(stand out) OD
        OD;
        x0:=x;
        ef1
      END;

   x:=(-1.2,1.0);
   dixon(f1,g1,2,1.0e-4,x)
END
FINISH
```

```
dixon

Enter Mouffti
d= +1.00000000e +0 +4.08163268e -1
x0= -1.20000000e +0 +1.00000000e +0
Initial amin=+7.7841535e+ 0        at point qoq=+2.5000000e- 1
x[ 1]=-5.0000000e- 1          p[ 1]=+4.4580779e+ 2
x[ 2]=-2.5000000e- 1          p[ 2]=+1.5109437e+ 2
x[ 3]=+0.0000000e+ 0          p[ 3]=+2.4200001e+ 1
x[ 4]=+2.5000000e- 1          p[ 4]=+7.7841535e+ 0
x[ 5]=+5.0000000e- 1          p[ 5]=+5.3881258e+ 1
a1=+1.7895e+ 3    b1=-3.0030e+ 2    c1=+6.3169e+ 1
d1=+1.0415e+ 0    e1=+2.6103e+ 0
its=    1      min=+1.7070961e- 1        qq=+4.1285026e+ 0
amin=+7.7841535e+ 0        qq=+4.1285026e+ 0
Leave Mouffti

h=
+1.4903231e- 1      -3.5518971e- 1
-3.5518971e- 1      +8.5181810e- 1
x=
-1.0292904e+ 0      +1.0696774e+ 0
ge=
+1.5684801e- 1      +2.0477355e+ 0
ef= +4.12850261e +0
p=
+2.1560001e+ 2      +8.8000003e+ 1



Enter Mouffti
d= +4.17357482e -1 -1.00000000e +0
x0= -1.02929039e +0 +1.06767740e +0
Initial amin=+3.3340089e+ 0        at point qoq=+2.5000000e- 1
x[ 1]=-5.0000000e- 1          p[ 1]=+5.1467198e+ 0
x[ 2]=-2.5000000e- 1          p[ 2]=+4.6719219e+ 0
x[ 3]=+0.0000000e+ 0          p[ 3]=+4.1285026e+ 0
x[ 4]=+2.5000000e- 1          p[ 4]=+3.8340089e+ 0
x[ 5]=+5.0000000e- 1          p[ 5]=+4.3905370e+ 0
a1=+7.0599e+ 0    b1=-2.5796e+ 0    c1=+2.5617e+ 0
d1=-4.7201e- 1    e1=+6.2043e- 1
its=    1      min=+2.3801370e- 1        qq=+3.8346130e+ 0
```



```
amin=+3.8340089e+ 0        qq=+3.8346130e+ 0
Leave Mouffti

h=
+4.2621360e- 2      -9.1741480e- 2
-9.1741476e- 2      +2.0025939e- 1
x=
-9.2995359e- 1      +8.3166369e- 1
ge=
-1.6191087e+ 1      -6.6299975e+ 0
ef= +3.83461297e +0
p=
+7.0463023e- 1      -1.6983134e+ 0



Enter Mouffti
d= +5.19044906e -1 -1.00000000e +0
x0= -9.29953590e -1 +8.31663691e -1
Initial amin=+3.5846060e+ 0        at point qoq=+2.5000000e- 1
x[ 1]=-5.0000000e- 1          p[ 1]=+5.4858556e+ 0
x[ 2]=-2.5000000e- 1          p[ 2]=+4.4132570e+ 0
x[ 3]=+0.0000000e+ 0          p[ 3]=+3.8346130e+ 0
x[ 4]=+2.5000000e- 1          p[ 4]=+3.5846060e+ 0
x[ 5]=+5.0000000e- 1          p[ 5]=+4.1783592e+ 0
a1=+1.0161e+ 0    b1=+1.7577e+ 0    c1=-2.9278e+ 0
d1=+2.9428e- 1    e1=-7.6352e- 1
its=    1      min=+2.7424004e- 1        qq=+3.5912805e+ 0
amin=+3.5846060e+ 0        qq=+3.5912805e+ 0
Leave Mouffti

ge1=
-2.3393698e+ 1      -1.2581392e+ 1
d=
+1.4234290e- 1      -2.7424004e- 1
y=
-7.2026112e+ 0      -5.9513942e+ 0
End of major step  h=
+4.3056244e- 2      -7.6025930e- 2
-7.6025927e- 2      +1.3808953e- 1
x=
-7.8761069e- 1      +5.5742365e- 1
ge=
-2.3393698e+ 1      -1.2581392e+ 1
ef= +3.59128055e +0
p=
+8.1840381e- 2      -1.5767495e- 1



Enter Mouffti
```

```
d= +1.00000000e +0 -8.11489098e -1
x0= -7.87610695e -1 +5.57423651e -1
Initial amin=+1.3533933e+ 0        at point goq=+7.5000000e- 1
x[ 1]=+0.0000000e+ 0        o[ 1]=+3.5912805e+ 0
x[ 2]=+5.0000000e- 1        o[ 2]=+2.1334783e+ 0
x[ 3]=+7.5000000e- 1        o[ 3]=+1.3533933e+ 0
x[ 4]=+8.7500000e- 1        o[ 4]=+3.4013838e+ 0
x[ 5]=+1.0000000e+ 0        o[ 5]=+9.5708785e+ 0
a1=+5.5701e+ 0   b1=-4.3486e+ 0   c1=+3.5066e+ 0
d1=-9.6839e- 1   e1=+9.7643e- 1
its=   1      min=+7.2262500e- 1        gq=+1.2444305e+ 0
amin=+1.3533933e+ 0        gq=+1.2444305e+ 0
Leave Mouffti

h=
+5.2643583e- 2     -5.9137767e- 2
-5.9137766e- 2     +1.0436701e- 1
x=
-6.4985692e- 2     -2.8978658e- 2
ge=
-2.9930281e+ 0     -6.6403596e+ 0
ef= +1.24443048e +0
p=
+5.0732791e- 2     -4.1169107e- 2



Enter Mouffti
d= -2.91217487e -1 +1.00000000e +0
x0= -6.49856916e -2 -2.89786579e -2
Initial amin=+1.2444305e+ 0        at point goq=+0.0000000e+ 0
x[ 1]=-5.0000000e- 1        o[ 1]=+2.9519002e+ 1
x[ 2]=-2.5000000e- 1        o[ 2]=+8.7707442e+ 0
x[ 3]=+0.0000000e+ 0        o[ 3]=+1.2444305e+ 0
x[ 4]=+2.5000000e- 1        o[ 4]=+5.3763901e+ 0
x[ 5]=+5.0000000e- 1        o[ 5]=+1.9670381e+ 1
a1=+3.9624e+ 3   b1=-1.1910e+ 2   c1=+5.3465e+ 1
d1=+3.8098e+ 0   e1=+4.2964e+ 1
its=   1      min=+3.1176262e- 2        gq=+1.1546962e+ 0
amin=+1.2444305e+ 0        gq=+1.1546962e+ 0
Leave Mouffti

h=
+2.9590791e- 2     -5.2146255e- 3
-5.2146255e- 3     +5.8625331e- 3
x=
-7.4064764e- 2     +2.1976042e- 3
ge=
-2.2455391e+ 0     -6.5759701e- 1
ef= +1.15469620e +0
p=
-1.2401992e- 1     +4.2586700e- 1



Enter Mouffti
```



```
d= +1.00000000e +0 -1.24633908e -1
x0= -7.40647642e -2 +2.19760422e -3
Initial amin=+1.0380649e+ 0        at point goq=+2.5000000e- 1
x[ 1]=-5.0000000e- 1        o[ 1]=+9.5019668e+ 0
x[ 2]=-2.5000000e- 1        o[ 2]=+2.2666753e+ 0
x[ 3]=+0.0000000e+ 0        o[ 3]=+1.1546962e+ 0
x[ 4]=+2.5000000e- 1        o[ 4]=+1.0380649e+ 0
x[ 5]=+5.0000000e- 1        o[ 5]=+6.1638164e+ 0
a1=-1.1049e+ 0   b1=+3.7709e- 1   c1=-3.7717e- 1
d1=+3.1811e- 2   e1=-3.2664e- 1
its=   1      min=+1.0601986e- 1        gq=+8.9717866e- 1
amin=+1.0380649e+ 0        gq=+8.9717866e- 1
Leave Mouffti

ge1=
-8.2479433e- 1     -5.3901061e+ 0
d=
+1.5601986e- 1     -2.0492395e- 2
y=
+1.4207447e+ 0     -4.7325391e+ 0
End of major step  h=
+1.1039773e- 1     -1.9382668e- 3
-1.9382664e- 3     +3.7905077e- 3
x=
+9.1955098e- 2     -1.8494791e- 2
ge=
-8.2479433e- 1     -5.3901061e+ 0
ef= +8.97178657e -1
p=
+6.3018154e- 2     -7.8544510e- 3



Enter Mouffti
d= +1.00000000e +0 +2.33631615e -1
x0= +9.19550983e -2 -1.84947907e -2
Initial amin=+8.9717866e- 1        at point goq=+0.0000000e+ 0
x[ 1]=-5.0000000e- 1        o[ 1]=+1.1091593e+ 1
x[ 2]=-2.5000000e- 1        o[ 2]=+2.3790395e+ 0
x[ 3]=+0.0000000e+ 0        o[ 3]=+8.9717866e- 1
x[ 4]=+2.5000000e- 1        o[ 4]=+1.0262339e+ 0
x[ 5]=+5.0000000e- 1        o[ 5]=+6.5214282e+ 0
a1=-3.2350e+ 0   b1=+4.5566e- 1   c1=-3.3647e- 1
d1=+1.9297e- 2   e1=-3.7504e- 1
its=   1      min=+1.1013568e- 1        gq=+7.4958400e- 1
amin=+8.9717866e- 1        gq=+7.4958400e- 1
Leave Mouffti

h=
+7.4488332e- 2     +2.6132306e- 2
+2.6132305e- 2     +1.6866923e- 2
x=
+2.0209078e- 1     +7.2363653e- 3
```

```
ge=
+1.1206289e+ 0      -6.7208593e+ 0
ef= +7.49584004e -1
p=
+8.0607959e- 2      +1.6832568e- 2



Enter Moufftí
d= +6.84043050e -1 +1.00000000e +0
x0= +2.02090776e -1 +7.23638531e -3
Initial amin=+7.4958400e- 1        at point qqq=+0.0000000e+ 0
x[ 1]=-5.0000000e- 1        p[ 1]=+2.7549103e+ 1
x[ 2]=-2.5000000e- 1        p[ 2]=+6.8792148e+ 0
x[ 3]=+0.0000000e+ 0        p[ 3]=+7.4958400e- 1
x[ 4]=+2.5000000e- 1        p[ 4]=+1.7861481e+ 0
x[ 5]=+5.0000000e- 1        p[ 5]=+4.6674563e+ 0
a1=+3.0382e+ 1   b1=-1.1945e+ 0   c1=+4.9242e- 1
d1=+5.6998e- 1   e1=+6.5692e- 1
its=   1      min=+5.2143936e- 2        qq=+5.8182330e- 1
amin=+7.4958400e- 1        qq=+5.8182330e- 1
Leave Moufftí

h=
+9.4751226e- 2      +4.2790343e- 2
+4.2790343e- 2      +2.4266913e- 2
x=
+2.3775947e- 1      +5.9380321e- 2
ge=
-1.7955985e+ 0      +5.7015075e- 1
ef= +5.81823304e -1
p=
+6.1864798e- 2      +9.0439919e- 2



Enter Moufftí
d= +1.00000000e +0 +4.32271514e -1
x0= +2.37759473e -1 +5.93803208e -2
Initial amin=+5.8182330e- 1        at point qqq=+0.0000000e+ 0
x[ 1]=-5.0000000e- 1        p[ 1]=+6.6794277e+ 0
x[ 2]=-2.5000000e- 1        p[ 2]=+1.2631309e+ 0
x[ 3]=+0.0000000e+ 0        p[ 3]=+5.8182330e- 1
x[ 4]=+2.5000000e- 1        p[ 4]=+7.5886710e- 1
x[ 5]=+5.0000000e- 1        p[ 5]=+7.2926605e+ 0
a1=-1.2598e+ 0   b1=+1.5746e- 1   c1=-1.8800e- 1
d1=+2.5748e- 2   e1=-3.2312e- 1
its=   1      min=+7.9375543e- 2        qq=+4.7104149e- 1
amin=+5.8182330e- 1        qq=+4.7104149e- 1
Leave Moufftí

ge1=
-4.9265599e- 1      -1.3765020e+ 0
d=
+7.9375541e- 2      +3.4311786e- 2
y=
```



```
+1.3029425e+ 0      -1.9466528e+ 0
End of major step  h=
+2.6059368e- 1      +1.3364636e- 1
+1.3364636e- 1      +7.1826747e- 2
x=
+3.1713501e- 1      +9.3692107e- 2
ge=
-4.9265599e- 1      -1.3765020e+ 0
ef= +4.71041486e -1
p=
+1.4573822e- 1      +6.2998479e- 2



Enter Moufftí
d= +1.00000000e +0 +5.27333595e -1
x0= +3.17135014e -1 +9.36921071e -2
Initial amin=+4.7104149e- 1        at point qqq=+0.0000000e+ 0
x[ 1]=-5.0000000e- 1        p[ 1]=+5.5369070e+ 0
x[ 2]=-2.5000000e- 1        p[ 2]=+1.0521257e+ 0
x[ 3]=+0.0000000e+ 0        p[ 3]=+4.7104149e- 1
x[ 4]=+2.5000000e- 1        p[ 4]=+1.1112125e+ 0
x[ 5]=+5.0000000e- 1        p[ 5]=+9.6651970e+ 0
a1=-2.4441e+ 0   b1=+1.7632e- 2   c1=-1.7036e- 1
d1=+3.2643e- 2   e1=-3.6156e- 1
its=   1      min=+7.7400858e- 4        qq=+4.7010045e- 1
amin=+4.7104149e- 1        qq=+4.7010045e- 1
Leave Moufftí

h=
+1.7188233e- 1      +1.0078884e- 1
+1.0078884e- 1      +6.1841934e- 2
x=
+3.1790902e- 1      +9.4100268e- 2
ge=
-4.7837573e- 1      -1.3931757e+ 0
ef= +4.70100451e -1
p=
+3.1234752e- 1      +1.6471134e- 1



Enter Moufftí
d= +1.00000000e +0 +8.56456973e -1
x0= +3.17909021e -1 +9.41002676e -2
Initial amin=+2.0716920e- 1        at point qqq=+2.5000000e- 1
x[ 1]=-5.0000000e- 1        p[ 1]=+1.4887192e+ 1
x[ 2]=-2.5000000e- 1        p[ 2]=+2.4219479e+ 0
x[ 3]=+0.0000000e+ 0        p[ 3]=+4.7010045e- 1
x[ 4]=+2.5000000e- 1        p[ 4]=+2.0716920e- 1
x[ 5]=+5.0000000e- 1        p[ 5]=+2.1836743e+ 0
```

```
a1=-4.0122e+ 0   b1=+7.4028e- 1    c1=-1.9774e- 1
d1=-4.0252e- 2   e1=-4.2064e- 1
its=   1     min=+1.7923979e- 1          qq=+2.5287997e- 1
amin=+2.0716920e- 1          qq=+2.5287997e- 1
Leave Mouffti

h=
+3.5317712e- 1      +2.6777620e- 1
+2.6777620e- 1      +2.1489382e- 1
x=
+4.9714881e- 1      +2.4761144e- 1
ge=
-1.0960830e+ 0      +9.0898946e- 2
ef= +2.52879974e -1
p=
+5.1547512e- 2      +4.4148226e- 2



Enter Mouffti
d= +1.00000000e +0 +7.55218625e -1
x0= +4.97148812e -1 +2.47611435e -1
Initial amin=+2.5287997e- 1          at point qog=+0.0000000e+ 0
x[ 1]=-5.0000000e- 1          p[ 1]=+2.6958667e+ 0
x[ 2]=-2.5000000e- 1          p[ 2]=+5.6730252e- 1
x[ 3]=+0.0000000e+ 0          p[ 3]=+2.5287997e- 1
x[ 4]=+2.5000000e- 1          p[ 4]=+1.5478293e+ 0
x[ 5]=+5.0000000e- 1          p[ 5]=+1.3622382e+ 1
a1=-4.9913e+ 0   b1=-2.9334e- 1    c1=-1.2957e- 1
d1=+1.3973e- 1   e1=-5.1238e- 1
its=   1     min=-6.2095222e- 2          qq=+3.3226174e- 1
amin=+2.5287997e- 1          qq=+3.3226174e- 1
Leave Mouffti

ge1=
-3.1214532e+ 0      +2.2888679e+ 0
d=
-6.2095221e- 2      -4.6895469e- 2
y=
-2.0253702e+ 0      +2.1979690e+ 0
End of major step   h=
+5.9940521e- 1      +5.2408485e- 1
+5.2408485e- 1      +4.6159450e- 1
x=
+4.3505359e- 1      +2.0071597e- 1
ge=
-3.1214532e+ 0      +2.2888679e+ 0
ef= +3.32261737e -1
p=
+3.6277086e- 1      +2.7397131e- 1
```



```
Enter Mouffti
d= +1.00000000e +0 +8.62869583e -1
x0= +4.35053591e -1 +2.00715967e -1
Initial amin=+3.3226174e- 1          at point qog=+0.0000000e+ 0
x[ 1]=-5.0000000e- 1          p[ 1]=+6.6536437e+ 0
x[ 2]=-2.5000000e- 1          p[ 2]=+9.0665708e- 1
x[ 3]=+0.0000000e+ 0          p[ 3]=+3.3226174e- 1
x[ 4]=+2.5000000e- 1          p[ 4]=+3.7866271e- 1
x[ 5]=+5.0000000e- 1          p[ 5]=+5.8690650e+ 0
a1=-8.8806e- 1   b1=+1.3761e- 1    c1=-9.9748e- 2
d1=+1.8832e- 2   e1=-3.0021e- 1
its=   1     min=+1.1302025e- 1          qq=+2.0469832e- 1
amin=+3.3226174e- 1          qq=+2.0469832e- 1
Leave Mouffti

h=
+5.4630417e- 1      +4.9870234e- 1
+4.9870234e- 1      +4.5732695e- 1
x=
+5.4807384e- 1      +2.9823770e- 1
ge=
-4.3311664e- 1      -4.2944551e- 1
ef= +2.04698317e -1
p=
+6.7145431e- 1      +5.7937750e- 1



Enter Mouffti
d= +1.00000000e +0 +9.88972269e -1
x0= +5.48073836e -1 +2.98237700e -1
Initial amin=+2.0469832e- 1          at point qog=+0.0000000e+ 0
x[ 1]=-5.0000000e- 1          p[ 1]=+4.8487520e+ 0
x[ 2]=-2.5000000e- 1          p[ 2]=+6.3598817e- 1
x[ 3]=+0.0000000e+ 0          p[ 3]=+2.0469832e- 1
x[ 4]=+2.5000000e- 1          p[ 4]=+8.7692128e- 1
x[ 5]=+5.0000000e- 1          p[ 5]=+9.3496969e+ 0
a1=-2.2980e+ 0   b1=-5.0688e- 2    c1=-7.1363e- 2
d1=+2.4124e- 2   e1=-3.4862e- 1
its=   1     min=-2.2223894e- 2          qq=+2.2482500e- 1
amin=+2.0469832e- 1          qq=+2.2482500e- 1
Leave Mouffti

h=
+3.5305985e- 1      +3.7186390e- 1
+3.7186391e- 1      +3.9545326e- 1
x=
+5.2584994e- 1      +2.7625889e- 1
ge=
-8.9376470e- 1      -5.1854551e- 2
ef= +2.24825002e -1
p=
+1.8579996e- 1      +1.8375101e- 1
```

```
Enter Mouffti
d= +9.44770418e -1 +1.00000000e +0
x0= +5.48073836e -1 +2.98237700e -1
Initial amin=+2.0469832e- 1          at point qog=+0.0000000e+ 0
x[ 1]=-5.0000000e- 1         o[ 1]=+5.1596059e+ 0
x[ 2]=-2.5000000e- 1         o[ 2]=+7.1392314e- 1
x[ 3]=+0.0000000e+ 0         o[ 3]=+2.0469832e- 1
x[ 4]=+2.5000000e- 1         o[ 4]=+4.9324804e- 1
x[ 5]=+5.0000000e- 1         o[ 5]=+5.9101286e+ 0
a1=-1.5382e+ 0   b1=+7.0825e- 2   c1=-6.8174e- 2
d1=+1.8427e- 2   e1=-3.3305e- 1
its=   1      min=+3.8727317e- 2          qq=+1.7487220e- 1
amin=+2.0469832e- 1          qq=+1.7487220e- 1
Leave Moufftî

ge1=
+3.0706291e- 1      -9.7298771e- 1
d=
+3.6588423e- 2      +3.8727317e- 2
y=
+7.4017955e- 1      -5.4354221e- 1
End of major step   h=
+2.6316528e- 1      +2.9105586e- 1
+2.9105586e- 1      +3.2510129e- 1
x=
+5.8466226e- 1      +3.3696502e- 1
ge=
+3.0706291e- 1      -9.7298771e- 1
ef= +1.74872203e -1
p=
+3.1261138e- 1      +3.3088607e- 1
```

```
Enter Moufftî
d= +8.91773812e -1 +1.00000000e +0
x0= +5.84662259e -1 +3.36965017e -1
Initial amin=+1.7487220e- 1          at point qog=+0.0000000e+ 0
x[ 1]=-5.0000000e- 1         o[ 1]=+4.0645029e+ 0
x[ 2]=-2.5000000e- 1         o[ 2]=+5.9990864e- 1
x[ 3]=+0.0000000e+ 0         o[ 3]=+1.7487220e- 1
x[ 4]=+2.5000000e- 1         o[ 4]=+4.6292810e- 1
x[ 5]=+5.0000000e- 1         o[ 5]=+5.0664291e+ 0
a1=-1.4473e+ 0   b1=+5.0017e- 2   c1=-6.0295e- 2
d1=+2.1353e- 2   e1=-3.4480e- 1
its=   1      min=+2.8650809e- 2          qq=+1.5648156e- 1
amin=+1.7487220e- 1          qq=+1.5648156e- 1
Leave Moufftî

h=
+2.0925375e- 1      +2.4352424e- 1
```



```
+2.4352426e- 1      +2.8629218e- 1
x=
+6.1021230e- 1      +3.6561583e- 1
ge=
+8.6634535e- 1      -1.3486460e+ 0
ef= +1.56481564e -1
p=
+2.0238548e- 1      +2.2694710e- 1
```

```
Enter Moufftî
d= +6.71678893e -1 +1.00000000e +0
x0= +6.10212304e -1 +3.65615826e -1
Initial amin=+5.9479743e- 2          at point qog=+2.5000000e- 1
x[ 1]=+0.0000000e+ 0         o[ 1]=+1.5648156e- 1
x[ 2]=+2.5000000e- 1         o[ 2]=+5.9479743e- 2
x[ 3]=+5.0000000e- 1         o[ 3]=+8.9335189e- 2
x[ 4]=+7.5000000e- 1         o[ 4]=+1.5834146e+ 0
x[ 5]=+1.0000000e+ 0         o[ 5]=+7.7872605e+ 0
a1=+8.2645e- 1   b1=-3.2980e- 1   c1=+1.4363e- 1
d1=-9.3902e- 1   e1=+9.1790e- 1
its=   1      min=+3.7496527e- 1          qq=+1.9691289e- 2
amin=+5.9479743e- 2          qq=+1.9691289e- 2
Leave Moufftî

h=
+2.7290655e- 1      +3.8523069e- 1
+3.8523068e- 1      +5.6714983e- 1
x=
+8.6206856e- 1      +7.4058110e- 1
ge=
+6.1417536e- 1      -5.1622242e- 1
ef= +1.96912894e -2
p=
+2.7572731e- 2      +4.1050466e- 2
```

```
Enter Moufftî
d= +5.56324631e -1 +1.00000000e +0
x0= +8.62068564e -1 +7.40581095e -1
Initial amin=+1.3736432e- 2          at point qog=+2.5000000e- 1
x[ 1]=-5.0000000e- 1         o[ 1]=+1.1804556e+ 0
x[ 2]=-2.5000000e- 1         o[ 2]=+1.7996772e- 1
x[ 3]=+0.0000000e+ 0         o[ 3]=+1.9691295e- 2
x[ 4]=+2.5000000e- 1         o[ 4]=+1.3736432e- 2
x[ 5]=+5.0000000e- 1         o[ 5]=+3.7422979e- 1
a1=-3.7092e- 1   b1=+5.2604e- 2   c1=-7.4488e- 3
d1=+1.1481e- 3   e1=-3.7828e- 1
its=   1      min=+1.4182968e- 1          qq=+4.3947569e- 3
amin=+1.3736432e- 2          qq=+4.3947569e- 3
Leave Moufftî

ge1=
+1.0176408e+ 0      -6.0347021e- 1
```

```
d=
+7.8903340e- 2      +1.4182968e- 1
y=
+4.0346545e- 1      -8.7247789e- 2
End of major step  h=
+3.2792776e- 1      +6.1209784e- 1
+6.1209782e- 1      +1.2049663e+ 0
x=
+9.4097190e- 1      +8.8241077e- 1
ge=
+1.0176408e+ 0      -6.0347021e- 1
ef= +4.39475692e -3
p=
+3.1252237e- 2      +5.6176260e- 2
```

```
Enter Moufft i
d= +3.42108473e -1 +1.00000000e +0
x0= +9.40971904e -1 +8.82410772e -1
Initial amin=+4.3947569e- 3          at point qoq=+0.0000000e+ 0
x[ 1]=-5.0000000e- 1          p[ 1]=+4.4781736e+ 0
x[ 2]=-2.5000000e- 1          p[ 2]=+1.0084356e+ 0
x[ 3]=+0.0000000e+ 0          p[ 3]=+4.3947569e- 3
x[ 4]=+2.5000000e- 1          p[ 4]=+6.2023713e- 1
x[ 5]=+5.0000000e- 1          p[ 5]=+2.1385657e+ 0
a1=+3.7874e+ 1   b1=-3.3365e- 1   c1=+1.3117e- 2
d1=+1.0429e+ 0   e1=+2.9847e+ 0
its=    1      min=+8.9034117e- 3          qq=+3.1360912e- 3
amin=+4.3947569e- 3          qq=+3.1360912e- 3
Leave Moufft i

h=
+3.9287765e- 1      +7.4051499e- 1
+7.4051498e- 1      +1.4007595e+ 0
x=
+9.4401783e- 1      +8.9131419e- 1
ge=
-1.6653289e- 1      +2.8902292e- 2
ef= +3.13609117e -3
p=
+3.5670128e- 2      +1.0426555e- 1
```

```
Enter Moufft i
d= +5.34020051e -1 +1.00000000e +0
x0= +9.44017835e -1 +8.91314186e -1
Initial amin=+2.9702132e- 3          at point qoq=+1.2500000e- 1
x[ 1]=-2.5000000e- 1          p[ 1]=+6.0294010e- 2
x[ 2]=-1.2500000e- 1          p[ 2]=+1.6139815e- 2
```



```
x[ 3]=+0.0000000e+ 0          p[ 3]=+3.1360912e- 3
x[ 4]=+1.2500000e- 1          p[ 4]=+2.9702132e- 3
x[ 5]=+2.5000000e- 1          p[ 5]=+4.4981612e- 2
a1=-4.3424e- 2   b1=+3.1052e- 3   c1=-3.6997e- 4
d1=+4.2855e- 2   e1=-1.1797e- 1
its=    1      min=+6.7984583e- 2          qq=+6.8798139e- 4
amin=+2.9702132e- 3          qq=+6.8798139e- 4
Leave Moufft i

h=
+4.1699757e- 1      +7.9921721e- 1
+7.9921723e- 1      +1.5360311e+ 0
x=
+9.8032296e- 1      +9.5929877e- 1
ge=
+6.4073367e- 1      -3.4686923e- 1
ef= +6.87981388e -4
p=
+4.3648246e- 2      +8.1735220e- 2
```

```
Enter Moufft i
d= +4.84611195e -1 +1.00000000e +0
x0= +9.80322964e -1 +9.59298767e -1
Initial amin=+6.8798139e- 4          at point qoq=+0.0000000e+ 0
x[ 1]=-2.5000000e- 1          p[ 1]=+1.0320737e- 1
x[ 2]=-1.2500000e- 1          p[ 2]=+1.9977880e- 2
x[ 3]=+0.0000000e+ 0          p[ 3]=+6.8798139e- 4
x[ 4]=+1.2500000e- 1          p[ 4]=+1.7411978e- 3
x[ 5]=+2.5000000e- 1          p[ 5]=+1.1857639e- 2
a1=-2.4185e+ 1   b1=+1.1961e+ 0   c1=-3.4697e- 2
d1=-5.9187e+ 1   e1=-5.0433e+ 1
its=    1      min=+4.8395628e- 2          qq=+1.5898726e- 5
amin=+6.8798139e- 4          qq=+1.5898726e- 5
Leave Moufft i

ge1=
-4.3871581e- 2      +2.5615096e- 2
d=
+2.3453064e- 2      +4.8395626e- 2
y=
-6.8460526e- 1      +3.7248433e- 1
End of major step  h=
+5.3959301e- 1      +1.0547055e+ 0
+1.0547055e+ 0      +2.0684161e+ 0
x=
+1.0037760e+ 0      +1.0076944e+ 0
ge=
-4.3871581e- 2      +2.5615096e- 2
ef= +1.58987259e -5
p=
+1.0039471e- 2      +2.0716548e- 2
```

```
4.2 condition set
x=
+1.0037760e+ 0     +1.0076944e+ 0
ge=
-4.3871581e- 2     +2.5615096e- 2
ef= +1.58987259e -5
p=
-3.3435849e- 3     -6.7110774e- 3
Enter Moufftl
d= -4.98218801e -1 -1.00000000e +0
x0= +1.00377603e +0 +1.00769439e +0
Initial amin=+1.5898726e- 5          at point gog=+0.0000000e+ 0
x[ 1]=-2.5000000e- 1          p[ 1]=+4.0295259e- 2
x[ 2]=-1.2500000e- 1          p[ 2]=+5.7884151e- 3
x[ 3]=+0.0000000e+ 0          p[ 3]=+1.5898726e- 5
x[ 4]=+1.2500000e- 1          p[ 4]=+4.8102441e- 3
x[ 5]=+2.5000000e- 1          p[ 5]=+3.8105999e- 2
a1=-3.4407e- 2   b1=+2.1139e- 4   c1=-1.8661e- 6
d1=+2.3260e- 3   e1=-1.1737e- 1
its=   1      min=+6.1426496e- 3          qq=+1.9510497e- 6
amin=+1.5898726e- 5       qq=+1.9510497e- 6
Leave Moufftl

h=
+5.1296348e- 1     +1.0273248e+ 0
+1.0273248e+ 0     +2.0657824e+ 0
x=
+1.0007156e+ 0     +1.0015517e+ 0
ge=
-4.6584791e- 2     +2.3990870e- 2
ef= +1.95104968e -6
p=
-3.3435849e- 3     -6.7110774e- 3



4.2 condition set
x=
+1.0007156e+ 0     +1.0015517e+ 0
ge=
-4.6584791e- 2     +2.3990870e- 2
ef= +1.95104968e -6
p=
+5.3461270e- 5     -8.9304012e- 5
Enter Moufftl
d= +5.98643541e -1 -1.00000000e +0
x0= +1.00071564e +0 +1.00155175e +0
Initial amin=+1.9510497e- 6          at point gog=+0.0000000e+ 0
x[ 1]=-2.5000000e- 1          p[ 1]=+2.7822237e+ 1
x[ 2]=-1.2500000e- 1          p[ 2]=+7.2571144e+ 0
x[ 3]=+0.0000000e+ 0          p[ 3]=+1.9510497e- 6
x[ 4]=+1.2500000e- 1          p[ 4]=+7.8595805e+ 0
x[ 5]=+2.5000000e- 1          p[ 5]=+3.2719784e+ 1
```



```
a1=+6.0535e+ 3   b1=-3.7598e- 1   c1=+2.4425e- 5
d1=-2.0480e+ 0   e1=+1.2519e+ 1
a1=-1.3928e+ 3   b1=+3.0391e- 1   c1=-5.6272e- 6   e1=-2.9842e+ 0
Quadratic Case

a1=+4.8373e+ 2   b1=+1.2049e+ 0   c1=+1.9510e- 6
its=   1      min=-2.4908980e- 3          qq=+3.1288434e- 3
amin=+1.9510497e- 6       qq=+3.1288434e- 3
Leave Moufftl

h=
+5.1440459e- 1     +1.0280170e+ 0
+1.0280170e+ 0     +2.0594486e+ 0
x=
+9.9922448e- 1     +1.0040426e+ 0
ge=
-2.2370436e+ 0     +1.1186138e+ 0
ef= +3.12884341e -3
p=
+5.3461270e- 5     -8.9304012e- 5



Trap 7 set

Enter Moufftl
d= +1.00000000e +0 -5.14993608e -1
x0= +1.00071564e +0 +1.00155175e +0
Initial amin=+1.9510497e- 6          at point gog=+0.0000000e+ 0
x[ 1]=-2.5000000e- 1          p[ 1]=+3.2179998e+ 1
x[ 2]=-1.2500000e- 1          p[ 2]=+8.9584211e+ 0
x[ 3]=+0.0000000e+ 0          p[ 3]=+1.9510497e- 6
x[ 4]=+1.2500000e- 1          p[ 4]=+1.0909644e+ 1
x[ 5]=+2.5000000e- 1          p[ 5]=+4.7878154e+ 1
a1=+1.3162e+ 3   b1=-2.2448e- 1   c1=+4.0492e- 6
d1=-8.4402e- 1   e1=+2.0754e+ 0
a1=-6.7838e+ 2   b1=+7.5516e- 1   c1=-2.1057e- 6   e1=-1.0793e+ 0
Quadratic Case

a1=+6.3578e+ 2   b1=+3.9024e+ 0   c1=+1.9510e- 6
its=   1      min=-6.1380629e- 3          qq=+2.4142047e- 2
amin=+1.9510497e- 6       qq=+2.4142047e- 2
Leave Moufftl

ge1=
-6.1884633e+ 0     +3.1056494e+ 0
d=
-6.1380640e- 3     +3.1610578e- 3
y=
-6.1418785e+ 0     +3.0816585e+ 0
End of major step  h=
+2.0193382e- 1     +4.0047101e- 1
+4.0047101e- 1     +7.9918190e- 1
x=
+9.9457758e- 1     +1.0047128e+ 0
ge=
-6.1884633e+ 0     +3.1056494e+ 0
ef= +2.41420474e -2
```

```
p=
+4.6584791e- 2     -2.3990870e- 2

Enter Mouffti
d= +1.00000000e +0 -6.19501628e -1
x0= +9.94577579e -1 +1.00471281e +0
Initial amin=+2.4142047e- 2        at point gog=+0.0000000e+ 0
x[ 1]=-2.5000000e- 1          p[ 1]=+3.6691030e+ 1
x[ 2]=-1.2500000e- 1          p[ 2]=+1.0643654e+ 1
x[ 3]=+0.0000000e+ 0          p[ 3]=+2.4142047e- 2
x[ 4]=+1.2500000e- 1          p[ 4]=+1.0653564e+ 1
x[ 5]=+2.5000000e- 1          p[ 5]=+4.8938928e+ 1
a1=+1.5265e+ 3   b1=-9.2335e+ 0   c1=+5.3821e- 2
d1=-8.7128e- 1   e1=+2.2294e+ 0
its=   1        min=+6.0494146e- 3          qq=+8.7568886e- 6
amin=+2.4142047e- 2         qq=+8.7568886e- 6
Leave Mouffti

h=
+2.0184855e- 1     +4.0041245e- 1
+4.0041245e- 1     +7.9928710e- 1
x=
+1.0006270e+ 0     +1.0009552e+ 0
ge=
+1.1700722e- 1     -5.7840347e- 2
ef= +8.75688863e -6
p=
+5.9375018e- 3     -3.6782920e- 3

4.2 condition set
x=
+1.0006270e+ 0     +1.0009652e+ 0
ge=
+1.1700722e- 1     -5.7840347e- 2
ef= +8.75688863e -6
p=
-3.4587770e- 4     -6.8940378e- 4
Enter Mouffti
d= -5.01705550e -1 -1.00000000e +0
x0= +1.00062700e +0 +1.00096518e +0
Initial amin=+8.7568886e- 6        at point gog=+0.0000000e+ 0
x[ 1]=-2.5000000e- 1          p[ 1]=+4.4895048e- 2
x[ 2]=-1.2500000e- 1          p[ 2]=+6.2465928e- 3
x[ 3]=+0.0000000e+ 0          p[ 3]=+8.7568886e- 6
x[ 4]=+1.2500000e- 1          p[ 4]=+5.2363970e- 3
x[ 5]=+2.5000000e- 1          p[ 5]=+3.8107670e- 2
a1=-3.8515e- 2   b1=+1.8300e- 4   c1=-1.0573e- 6
```



```
d1=-5.1150e- 3   e1=-1.2074e- 1
its=   1        min=+4.7507691e- 3          qq=+1.0685964e- 5
amin=+8.7568886e- 6         qq=+1.0685964e- 5
Leave Mouffti

h=
+4.7325948e- 1     +9.4597329e- 1
+9.4597329e- 1     +1.8958369e+ 0
x=
+9.9824351e- 1     +9.9621441e- 1
ge=
+1.0657085e- 1     -5.5138767e- 2
ef= +1.06859636e -5
p=
-3.4587770e- 4     -6.8940378e- 4

Trap 7 set

Enter Mouffti
d= -1.00000000e +0 +4.94331446e -1
x0= +1.00062700e +0 +1.00096518e +0
Initial amin=+8.7568886e- 6        at point gog=+0.0000000e+ 0
x[ 1]=-1.2500000e- 1          p[ 1]=+1.0765157e+ 1
x[ 2]=-6.2500000e- 2          p[ 2]=+2.5694135e+ 0
x[ 3]=+0.0000000e+ 0          p[ 3]=+8.7568886e- 6
x[ 4]=+6.2500000e- 2          p[ 4]=+2.3075033e+ 0
x[ 5]=+1.2500000e- 1          p[ 5]=+8.7790803e+ 0
a1=+1.2930e+ 3   b1=-1.4078e- 1   c1=+1.8153e- 5
d1=+8.3459e- 1   e1=+2.0730e+ 0
its=   1        min=+1.0587805e- 4          qq=+2.9904534e- 7
amin=+8.7568886e- 6         qq=+2.9904534e- 7
Leave Mouffti

ge1=
+8.0374376e- 3     -3.4987926e- 3
d=
-1.0888278e- 4     +5.3822994e- 5
y=
-1.0896978e- 1     +5.4341555e- 2
End of major step   h=
+1.9996243e- 1     +3.9897605e- 1
+3.9897605e- 1     +8.0104730e- 1
x=
+1.0005181e+ 0     +1.0010190e+ 0
ge=
+8.0374376e- 3     -3.4987926e- 3
ef= +2.99045343e -7
p=
-1.1700722e- 1     +5.7840347e- 2
```

(vii) Finally a special method due to Pekam[14] is used to deal with functions which are sums of squares.


```
PROGRAM pekam
 BEGIN
   INT n;
  n:=2;
  [1:n] PROC ( REF [ ] REAL ) REAL f;
  [1:n] REAL x0,de;
  read((x0,de));

   PROC gaussp=( INT n, REF [,] REAL a, REF [ ] REAL b,x) VOID :
   BEGIN
      INT ml;
      REAL max,w;
      FOR r TO n-1 DO
      BEGIN
        max:= ABS a[r,r];
        ml:=r;
         FOR i1 FROM r+1 TO n DO
         IF  ABS a[i1,r]>max THEN max:= ABS a[i1,r];ml:=i1 FI  OD ;
         IF ml/=r THEN
            FOR i1 FROM r TO n DO
          (w:=a[ml,i1];a[ml,i1]:=a[r,i1];a[r,i1]:=w) OD ;
          w:=b[ml];b[ml]:=b[r];b[r]:=w
         FI ;
         FOR i1 FROM r+1 TO n DO
        (w:=-a[i1,r]/a[r,r];
          FOR i2 FROM r TO n DO a[i1,i2] PLUSAB w*a[r,i2] OD ;
         b[i1] PLUSAB w*b[r]) OD
      END  OD ;
    x[n]:=b[n]/a[n,n];
     FOR i1 FROM n-1 BY -1 TO 1 DO
    (x[i1]:=b[i1];
      FOR i2 FROM i1+1 TO n DO x[i1] MINUSAB x[i2]*a[i1,i2] OD ;
     x[i1] DIVAB a[i1,i1]) OD
   END ;

   PROC mm=( REF [,] REAL a,b,c) VOID :
   BEGIN
      INT n:= UPB a,m1:=2 UPB a,m2:= UPB b,p:=2 UPB b;
      IF m1/=m2 OR  UPB c/=n OR 2 UPB c/=p THEN
        print((newline,"bounds mismatched"))
      ELSE
         FOR i1 TO n DO
         FOR i2 TO p DO
        ( REAL s:=0.0;
         FOR i3 TO m1 DO s PLUSAB a[i1,i3]*b[i3,i2] OD ;
         c[i1,i2]:=s) OD
      OD  FI
   END ;

   PROC tr=( REF [,] REAL a,b) VOID :
   BEGIN
      INT n:= UPB a,m:=2 UPB b,m1:=2 UPB a,m2:= UPB b;
      IF m1/=m2 OR n/=m
      THEN print((newline,"tr mismatch bounds"))
```

```
   ELSE
      FOR i1 Tu n DJ  FuR i2 TO m1 DO b[i2,i1]:=a[i1,i2]
   OD  OD  FI
END ;

PROC mv=( REF [,] REAL a, REF [ ] REAL v,b) VOID :
BEGIN
   INT n1:= UPB a,n2:=2 UPB a,n3:= UPB v,n4:= UPB b;
   IF n2/=n3 OR n1/=n4
   THEN print((newline,"mv mismatch"))
   ELSE
      FOR i1 TO n1 DO
     ( REAL s:=0.0;
       FuR i2 TO n2 Du s PLUSAB a[i1,i2]*v[i2] OD ;
      b[i1]:=s) OD
   FI
END ;

PROC fout=( REF [,] REAL x, REF [ ] REAL sl) VOID :
BEGIN
   INT n1:= UPB x,n2:=2 UPB x,n3:= UPB sl;
   FuRMAT ft1=$l5(+d.9de+zd3x)$,
          ft2=$l"x["7zd"]"l$;
   FOR i1 TO n2 DO
  (printf((ft2,i1));
  printf((ft1,x[ ,i1]))) OD;
  CO
  printf((ft1,sl));
  CO
    TO 4 DO newline(stand out) OD
END ;

PROC pekam=([ ] PROC ( REF [ ] REAL ) REAL f, REF [ ] REAL x0,de,
               INT n,m, REAL e) VOID :
BEGIN
   INT p=n+3+n%3;
  [1:n,1:p] REAL x;
  [1:m,1:p] REAL fa;
  [1:n] REAL xbar;
  [1:p] REAL w,ws,sl;
   FORMAT ft1=$l+d.8de+zd$,
             ft2=$l+d.9de+zd,4(3x+d.9de+zd)$,
            ft3=$l"accepted new point is x[ ,"6zd"]"l$;
   REAL smax,smin,s2;
   INT mi;

   PROC newx=( INT pp, BOOL replace) VOID :
   BEGIN
     [1:n,1:pp] REAL xp,ar2;
     [1:m,1:pp] REAL fp;
     [1:pp,1:n] REAL xpt;
     [1:pp,1:m] REAL fpt;
     [1:n,1:m] REAL ar1;
     [1:n,1:n] REAL ar;
     [1:n] REAL b,z,nx;
      REAL s1,s2,w2,w3;
      INT ml;
     [1:m] REAL newf;
     smax:=sl[1];ml:=1;
     FuR i1 FRuM 2 TJ pp Du
```



```
       IF sl[i1]>smax THEN
          smax:=sl[i1];ml:=i1
       FI
   OD;
   s1:=0.0;
    FOR i1 TO pp DO s1 PLUSAB ws[i1] OD ;
    FOR i1 TO n DO
   (s2:=0.0;
     FOR i2 TU pp DU s2 PLUSAB ws[i2]*x[i1,i2] OD ;
    xbar[i1]:=s2/s1) OD ;
   FOR i1 TO n DO FOR i2 TO pp DO
      xp[i1,i2]:=w[i2]*(x[i1,i2]-xbar[i1])
   OD OD;
    FOR i1 TO m DO  FOR i2 TO pp DO fp[i1,i2]:=w[i2]*fa[i1,i2]
   OD OD;
   tr(xp,xpt);
   tr(fp,fpt);
   mm(xp,fpt,ar1);
   mm(ar1,fp,ar2);
   mm(ar2,xpt,ar);
   mv(ar2,w[1:pp],b);
   gaussp(n,ar,b,z);
   mm(xp,xpt,ar);
   mv(ar,z,nx);
    FOR i1 TO n DO nx[i1]:=-nx[i1]/s1+xbar[i1] OD ;
   s2:=smax+1.0;
    TO 10 WHILE s2>smax DO
    BEGIN
    FOR i1 TO m DO newf[i1]:=f[i1](nx) OD ;
   print((newline,"new point is",newline));
   printf((ft2,nx));
   print((newline,"value of m functions at new point ",newline));
   printf((ft2,newf));
   s2:=0.0;
    FOR i1 TO m DO s2 PLUSAB newf[i1]*newf[i1] OD ;
    IF s2<e*e THEN x0:=nx; GOTO fin FI ;
   w3:=1.0/s2;
   w2:=sqrt(w3);
    IF s2>smax THEN
       REAL min; INT mn;
      print((newline,"unacceptable point",newline));
      min:=sl[1];mn:=1;
       FOR i1 FROM 2 TO pp DO  IF sl[i1]<min THEN min:=sl[i1];mn:=i1
      FI OD;
       FOR i1 TO n DO
         nx[i1]:=(w2*(nx[i1]-xbar[i1])+w[mn]*(x[i1,mn]-xbar[i1]))
                 /(w2+w[mn])+xbar[i1]
      OD
    FI
    END  OD ;
    IF replace THEN  SKIP  ELSE ml:=pp+1 FI ;
    FOR i1 TO n DO x[i1,ml]:=nx[i1] OD ;
    FOR i1 TO m DO fa[i1,ml]:=f[i1](x[ ,ml]) OD ;
   s2:=0.0;
    FOR i1 TO m DO s2 PLUSAB fa[i1,ml]*fa[i1,ml] OD ;
   sl[ml]:=s2;
   printf((ft3,ml));
   print((newline,"x-values"));
   printf((ft2,x[ ,ml]));
   print((newline,"f-values"));
```

```
          printf((ft2,fa[ ,ml]));
          print((newline,"s2=",s2,newline));
          ws[ml]:=1.0/s2;
          w[ml]:=sqrt(ws[ml])
        END ;

      x[ ,1]:=x0;
       FOR i1 FROM 2 TO n+1 DO
         (x[ ,i1]:=x0;
          x[i1-1,i1] PLUSAB de[i1-1]) OD ;
       FOR i1 TO n+1 DO  FOR i2 TO m DO fa[i2,i1]:=f[i2](x[ ,i1])
       OD OD;
      printf((ft1,x0));
       FOR i1 TO n+1 DO
         (s2:=0.0;
           FOR i2 TO m DO s2 PLUSAB fa[i2,i1]*fa[i2,i1] OD ;
          sl[i1]:=s2;
          ws[i1]:=1.0/s2;
          w[i1]:=sqrt(ws[i1])) OD ;
       FOR i1 FROM n+2 TO p DO
          newx(i1-1, FALSE ) OD ;
       FOR ia
       WHILE
         ia<50 OR
         (smin:=sl[1];mi:=1;
           FOR i1 TO p DO  IF sl[i1]<smin THEN smin:=sl[i1];mi:=i1  FI
          OD;
          smin>e*e)
       DO
       BEGIN
         fout(x,sl);
         newx(p, TRUE )
       END  OD ;
      x0:=x[ ,mi];
   fin: SKIP
    END ;


   f[1]:=( REF [ ] REAL x) REAL :
    BEGIN
       REAL c1:=cos(x[1]),c2:=cos(x[2]),w:=x[1]-x[2];
       IF  ABS w<=1.0e-20 THEN
       REAL s1:=sin(x[2]);
      (x[2]*x[2]-pi*pi/4)*(-s1-w*c2/2)/2
      +(x[2]-pi/2)*(c2+w*x[2]*c2/2+x[2]*s1)+1-s1
       ELSE
      (x[1]*x[1]-pi*pi/4)*(c1-c2)/2/w
       +(x[1]-pi/2)*(x[1]*c2-x[2]*c1)/w+1-sin(x[1])
       FI
    END ;
     f[2]:=( REF [ ] REAL x) REAL :
      BEGIN
         REAL c1:=cos(x[1]),c2:=cos(x[2]),w:=x[1]-x[2];
         IF  ABS w<=1.0e-20 THEN
         REAL s1:=sin(x[2]);
        x[2]*x[2]*(-s1-w*c2/2)/2+x[2]*(c2+w*x[2]*c2/2+x[2]*s1)-s1
         ELSE
        x[2]*x[2]*(c1-c2)/2/w+x[2]*(x[1]*c2-x[2]*c1)/w-sin(x[2])
         FI
      END ;

    pekam(f,x0,de,n,n,1.0e-07);
     FOR i1 TO n DO
       printf(($l"x["6zd,"]"+d.9de+zd$,i1,x0[i1])) OD
  END
  FINISH
```

```
pekam
0.0      0.8
0.5   0.5

+0.00000000e+ 0
+7.99999997e- 1
new point is

+4.981907420e- 1   +9.617869630e- 1
value of m functions at new point

-3.988553956e- 2   +3.566022217e- 2
accepted new point is x[ ,       4]

x-values
+4.981907420e- 1   +9.617869630e- 1
f-values
-3.988553956e- 2   +3.566022217e- 2
s2= +2.86250771e -3

new point is

+4.079880677e- 1   +1.048247248e+ 0
value of m functions at new point

-2.187873051e- 2   +1.602199674e- 2
accepted new point is x[ ,       5]

x-values
+4.079880677e- 1   +1.048247248e+ 0
f-values
-2.187873051e- 2   +1.602199674e- 2
s2= +7.35383226e -4

x[        1]

+0.000000000e+ 0   +7.999999970e- 1
x[        2]

+5.000000000e- 1   +7.999999970e- 1
x[        3]

+0.000000000e+ 0   +1.299999997e+ 0
x[        4]
```



```
+4.981907420e- 1   +9.617869630e- 1
x[        5]

+4.079880677e- 1   +1.048247248e+ 0


new point is

+3.614973091e- 1   +1.111135334e+ 0
value of m functions at new point

-5.184184760e- 3   +1.670107245e- 3
accepted new point is x[ ,       3]

x-values
+3.614973091e- 1   +1.111135334e+ 0
f-values
-5.184184760e- 3   +1.670107245e- 3
s2= +2.96650298e -5

x[        1]

+0.000000000e+ 0   +7.999999970e- 1
x[        2]

+5.000000000e- 1   +7.999999970e- 1
x[        3]

+3.614973091e- 1   +1.111135334e+ 0
x[        4]

+4.981907420e- 1   +9.617869630e- 1
x[        5]

+4.079880677e- 1   +1.048247248e+ 0


new point is

+3.578691706e- 1   +1.132003501e+ 0
value of m functions at new point

-4.190951586e- 5   -7.445737720e- 4
accepted new point is x[ ,       1]

x-values
+3.578691706e- 1   +1.132003501e+ 0
f-values
-4.190951586e- 5   -7.445737720e- 4
s2= +5.56146510e -7

x[        1]

+3.578691706e- 1   +1.132003501e+ 0
x[        2]
```

```
+5.000000000e- 1   +7.999999970e- 1
x[        3]

+3.614973091e- 1   +1.111135334e+ 0
x[        4]

+4.981907420e- 1   +9.617869630e- 1
x[        5]

+4.079880677e- 1   +1.048247248e+ 0



new point is

+3.621690311e- 1   +1.132126495e+ 0
value of m functions at new point

-3.615505993e- 4   +3.516823053e- 4
accepted new point is x[ ,      2]

x-values
+3.621690311e- 1   +1.132126495e+ 0
f-values
-3.615505993e- 4   +3.516823053e- 4
s2= +2.54399282e -7

x[        1]

+3.578691706e- 1   +1.132003501e+ 0
x[        2]

+3.621690311e- 1   +1.132126495e+ 0
x[        3]

+3.614973091e- 1   +1.111135334e+ 0
x[        4]

+4.981907420e- 1   +9.617869630e- 1
x[        5]

+4.079880677e- 1   +1.048247248e+ 0



new point is

+3.610395193e- 1   +1.133403003e+ 0
value of m functions at new point

+2.337247133e- 5   -3.197044134e- 5
accepted new point is x[ ,      4]

x-values
+3.610395193e- 1   +1.133403003e+ 0
f-values
+2.337247133e- 5   -3.197044134e- 5
```



```
s2= +1.56838154e -9

x[        1]

+3.578691706e- 1   +1.132003501e+ 0
x[        2]

+3.621690311e- 1   +1.132126495e+ 0
x[        3]

+3.614973091e- 1   +1.111135334e+ 0
x[        4]

+3.610395193e- 1   +1.133403003e+ 0
x[        5]

+4.079880677e- 1   +1.048247248e+ 0



new point is

+3.611487933e- 1   +1.133337528e+ 0
value of m functions at new point

-5.923211575e- 7   +9.760260582e- 7
accepted new point is x[ ,      5]

x-values
+3.611487933e- 1   +1.133337528e+ 0
f-values
-5.923211575e- 7   +9.760260582e- 7
s2= +1.30347122e-12

x[        1]

+3.578691706e- 1   +1.132003501e+ 0
x[        2]

+3.621690311e- 1   +1.132126495e+ 0
x[        3]

+3.614973091e- 1   +1.111135334e+ 0
x[        4]

+3.610395193e- 1   +1.133403003e+ 0
x[        5]

+3.611487933e- 1   +1.133337528e+ 0



new point is

+3.611453846e- 1   +1.133338913e+ 0
value of m functions at new point

+7.450580597e- 9   -7.450580597e- 9

x[        1]+3.611453846e- 1
x[        2]+1.133338913e+ 0
```

Reading List

(1) Gill, P.E., Murray, W. and Wright, M.H., "Practical Optimization" Academic Press 1981.

(2) Murray, W. (ed.), "Numerical Methods for Unconstrained Optimization" Academic Press, 1972.

(3) Gill, P.E. and Murray, W. "Numerical Methods for Constrained Optimization" Academic Press, 1974.

(4) Fletcher, R., "Practical Methods of Optimization", Vol. I and II, John Wiley and Sons, Chichester 1981.

References

[1] Walsh, G.A., "Methods of Optimization", John Wiley and Sons 1975.

[2] Powell, M.J.D., "An efficient method for finding stationary values of a function without calculating derivatives", C.J. 7, 303-307, 1965.

[3] Davidon, W.C., "Variable metric method for minization", A.E.C. R and D. Report ANL-5990 (rev) Argonne National Laboratory 1959.

[4] Broyden, C.G., "Quasi-Newton Methods and their application to function Minimization", Maths. Comput. 21, 369-381.

[5] Rosenbrock, H.H., "An automatic method for finding the greatest or least value of a function", C.J. 3, 175-184, 1960.

[6] Nelder, J.A. and Mead, R., "A simplex method for function minimization", C.J. 7, 308-313, 1965.

[7] Gill, P.E. and Murray, W., "Newton-type methods for unconstrained and linearly constrained optimization", Maths. Prog. 28, 311-350, 1974.

[8] Fletcher, R. and Powell, M.J.D., "A rapidly convergent descent method for minimization", C.J. 6, 163-168, 1963.

[9] Dixon, L.C.W., "Quasi-Newton algorithms generate identical points", Maths. Prog. 2, 383-387, 1972.

[10] Huang, H.Y., "Unified approach to quadratically convergent algorithms for function minimization", J. of Opt. Th. and App. 5, 405-423, 1970.

[11] Fiacco, A.V. and McCormick, G.P., "Non-linear programming: Sequential unconstrained minimization techniques", John Wiley and Sons, 1968.

[12] Powell, M.J.D., "A method for non-linear constraints in minimization problems in optimization", (R. Fletcher, Ed.) 283-298, Academic Press 1969.

[13] Dixon, L.C.W. "Conjugate directions without linear searches", J. Inst. Maths. Applics. (1973) 11, 317-328.

[14] Peckam, G., "A new method for minimizing a sum of squares without calculating gradients", C.J. 13, 4, (1970), 418-420.

[15] Winfield, D., "Function minimization by interpolation in a data table, J. Inst. Maths. Applics. 12, (1973), 339-347.

[16] Storey, C., "Optimization using rational functions" Mathematics Research 131, Loughborough University of Technology.

[17] Hyslop, J., "A note on the accuracy of optimization techniques", C.J. 15, (1972), 140.