

Hello everyone,  
  
The following contains the  
SIMPLEX METHOD (Linear Programming, Maximization and Minimization)  
  
Using material from  
Frederick S. Hiller, Gerald J. Lieberman. Introduction to Operations Research. New York: McGraw Hill, 2005. Print.  
  
Please let us know, if you would like us to include modifications, improvements, and any additions to the code.   
  
  
Truly,  
  
Haulwind Team

=============================================================

**maximize Z,**

where Z = C \* X AND A \* X ≤ b

where C,X, and b are vectors; and A is a Matrix

e.g. max Z; C1X1 + C2X2 = Z

A1X1+ A2X2 ≤ b1

A3X1+ A4X2 ≤ b2

A5X1+ A5X2 ≤ b3

**minimize Z,**

where Z = b \* Y AND y \* A ≥ C

where C,Y, and b are vectors; and A is a Matrix

e.g. max Z; b1Y1 + b2Y2 + b3Y3 = Z

A1Y1+ A3Y2+ A5Y3 ≥ C1

A2Y1+ A4Y2+ A6Y3 ≥ C2

=============================================================

**MAXIMIZATION INPUT**

C = {C1, C2}= L1

b = {b1, b2,b3}= L2

A = [ A1 A2

A3 A4

A5 A6 ]

= [A]

OUTPUT

X = [X1 X2]= LE

Z = Z

**MINIMIZATION INPUT**

C = {C1, C2}= L2

b = {b1, b2,b3}= L1

A = [ A1 A3 A5

A2 A4 A6 ]

= [A]

OUTPUT

Y = [Y1 Y2 Y3]= LF

Z = Z

=============================================================

DIM(L1) -> N

DIM(L2) -> M

{1,1} -> DIM[B]

0 -> DIM(L5)

DISP "CHOOSE 1 FOR MAXIMIZATION SETUP OR

2 FOR MINIMIZATION SETUP "

INPUT X

IF X = 1

THEN GOTO 2

ELSE

FOR (A,1,N,1)

L1(A) -> L5(A)

END

0 -> DIM(L1)

FOR (B,1,M,1)

L2(B) -> L1(B)

END

0 -> DIM(L2)

FOR (C,1,N,1)

L5(C) -> L2(C)

END

{N,M} -> DIM([B])

FOR (C,1,M,1)

FOR (D,1,N,1)

[A](C,D) -> [B](D,C)

END END

{1,1} -> DIM([A])

[B]->[A]

END

LBL 2

DIM(L1) -> N

DIM(L2) -> M

-L1 -> L1

AUGMENT([A],IDENTITY[M]) -> [C]

1 -> K

WHILE L1(K) ≠ MIN(L1)

K+1 -> K

END

0 -> DIM(L4)

M -> DIM(L4)

0 -> DIM(L3)

0 -> DIM(L5)

0 -> DIM(LZ)

{1,1} -> DIM([B])

L2 -> LD

L1 -> LX

FOR (A,1,M,1)

A+N -> L3(A)

END

LBL 1

AUGMENT (-LX,L4) -> L6

1 -> P

0 -> DIM(LK)

FOR(Q,1,M,1)

IF [A](Q,K) ≠ 0

THEN

L2(Q)/[A](Q,K) -> LK(Q)

ELSE

MAX(L2) -> LK(Q)

END END

1 -> Y

FOR (Q,1,M,1)

IF [A](Q,K) ≠ 0

THEN

IF (L2(Q)/[A](Q,K)) = MIN(LK)

THEN

IF Y = 1

K -> L3(Q)

Y+1 -> Y

END END END END

{M,M} -> DIM([B])

FOR(A,1,M,1)

FOR(B,1,M,1)

[C](B,L3(A)) -> [B](B,A)

END END

[B]-1 -> [D]

FOR (A,1,M,1)

L6(L3(A)) -> L5(A)

END

L5 -> LZ

0 -> DIM(LC)

FOR(A,1,M,1)

FOR(B,1,M,1)

[D](A,B)\*LD(B) -> L5(B)

END

SUM(L5) -> LC(A)

END

LC -> L2

N -> DIM(LE)

0 -> DIM(LF)

FOR(B,1,M,1)

FOR(A,1,M,1)

LZ(A)\*[D](A,B) -> L5(A)

END

SUM(L5) -> LF(B)

END

N -> DIM(LY)

FOR(B,1,N,1)

FOR(A,1,M,1)

LF(A)\*[A](A,B) -> L5(A)

END

SUM(L5) -> LY(B)

END

LY + LX -> L1

IF MIN(L1)<0

THEN

1 -> K

WHILE L1(K) ≠ MIN (L1)

K+1 -> K

END

GOTO 1

END

SUM (LZ\*L2) -> Z

DISP ''THE OPTIMAL SOLUTION IS''

DISP Z

IF X=2

THEN

DISP ''THE OPTIMAL QUANTITY FOR EACH RESOURCE IS''

DISP LF

ELSE

FOR(A,1,M,1)

FOR(B,1,N,1)

IF L1(B) = 0

THEN

IF L3(A) = B

THEN

L2(A) -> LE(B)

END

ELSE

0 -> LE(B)

END END END

DISP ''THE OPTIMAL QUANTITY FOR EACH RESOURCE IS''

DISP LE

END

0 -> DIM(L5)

0 -> DIM(L6)

IF X = 1

THEN

FOR(B,1,M,1)

FOR(A,1,N,1)

LE(A)\*[A](B,A) -> L5(A)

END

SUM(L5) -> L6(B)

END

LD-L6 -> L6

0 -> DIM(L5)

DISP "EXCESS RESOURCES: FIRST VALUE IS VECOR IS THE LINE, THE SECOND VALUE IS THE EXCESS AMOUNT"

FOR(A,1,M,1)

IF L6(A) > 0

THEN

A -> L5(1)

L6(A) -> L5(2)

DISP L5

END END

ELSE

FOR(A,1,N,1)

FOR(B,1,M,1)

LF(B)\*[A](B,A) -> L5(B)

END

SUM(L5) -> L6(A)

END

L6+LX -> L6

0 -> DIM(L5)

DISP "EXCESS RESOURCES: FIRST VALUE IS VECOR IS THE LINE, THE SECOND VALUE IS THE EXCESS AMOUNT"

FOR (A,1,N,1)

IF L6(A) > 0

THEN

A -> L5(1)

L6(A) -> L5(2)

DISP L5

END END END

CLEARALLLISTS

=============================================================