

BAB III.

PERUBAHAN PADA VEKTOR C

Sesudah memperoleh SFB \bar{x}^* yang optimum pada persoalan semula, mungkin saja terjadi perubahan pada koefisien fungsi tujuan, karena adanya perubahan keuntungan / ongkos baik pada kegiatan basis maupun nonbasis. Bab ini akan memperlihatkan bagaimana \bar{x}^* pada persoalan semula dipakai untuk memecahkan persoalan baru yang mana koefisien fungsi tujuannya dirubah sedikit atau bahkan berubah total. Oleh sebab itu SFB \bar{x}^* yang optimum pada persoalan semula dites, apakah juga memenuhi syarat optimalitas pada fungsi tujuan yang baru, pada persoalan :

minimumkan : $\bar{c}\bar{x}$

dengan pembatas : $A\bar{x} = \bar{b}$

$\bar{x} \geq 0$

III.1. Sensitivitas dari koefisien fungsi tujuan.

Fungsi : untuk mengetahui sejauh mana koefisien fungsi tujuan (\bar{c}) itu dapat dirubah, tetapi solusi optimumnya tetap \bar{x}^* pada persoalan semula.

Agar \bar{x}^* tetap optimal untuk fungsi tujuan yang manapun juga, harus dipenuhi $\bar{c}_N - \bar{c}_B B^{-1} N \geq 0$ (karena, syarat untuk optimal adalah harga reduksi ≥ 0).

Untuk menentukan daerah dari koefisien fungsi tujuan (\bar{c}) agar dipenuhi \bar{x}^* tetap optimal, timbul 2 persoalan :

1. Jika yang berubah adalah koefisien fungsi tujuan nonbasis

(\bar{c}_N), dalam hal ini koefisien fungsi tujuan basis (\bar{c}_B)

tidak berubah (tetap seperti pada persoalan semula).

Oleh karenanya $\bar{c}_B B^{-1} N$ juga tidak berubah.

Mengingat syarat optimalitas adalah $\bar{c}_N - \bar{c}_B B^{-1} N \geq 0$

maka : sejauh $(c_N)_j - (c_B B^{-1} N)_j \geq 0$

atau : untuk setiap c_N yang memenuhi $(c_N)_j \geq (c_B B^{-1} N)_j$,

\bar{x}^* adalah tetap optimal.

2. Jika yang berubah adalah koefisien fungsi tujuan basis $(c_B)_k$ dalam hal ini koefisien fungsi tujuan nonbasis (\bar{c}_N) tidak berubah (tetap seperti pada persoalan semula).

Dengan juga mengingat syarat optimalitas, maka $(c_B)_k$ akan dibatasi oleh batasan - batasan tertentu $t_1 \leq (c_B)_k \leq t_2$ agar \bar{x}^* tetap optimal.

III.2. Parametrik dari fungsi tujuan.

Fungsi : untuk menentukan masing - masing nilai optimal dari setiap $s \geq 0$ yang memenuhi PL parametrik di bawah ini.

minimumkan $(\bar{c} + s\bar{c}^1) x$

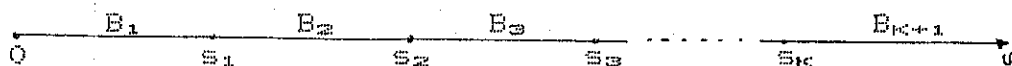
pembatas $A\bar{x} = \bar{b}$

$$\bar{x} \geq 0$$

$$s \geq 0$$

Karena nilai dari s adalah tak berhingga banyaknya, sehingga ada tak berhingga permasalahan yang harus diselesaikan.

Sedangkan SFB yang memenuhi pembatas - pembatasnya hanyalah berhingga banyaknya, maka $0 < s_1 < s_2 < \dots < s_k < s$ dibagi menjadi berhingga banyak interval



yang mana setiap interval $[s_{i-1}, s_i]$ mempunyai matrik basis $B_i, \forall i$ dengan nilai optimalnya adalah $\bar{x}_B = B_i^{-1}b$.

(<http://eprints.undip.ac.id>)

Akan diberikan beberapa algoritma untuk memperoleh nilai dari s_1, s_2, \dots, s_k beserta basis B_1, \dots, B_{k+1} dan solusi optimal pada selang interval tertentu.

Langkah 0 :

Ambil $s=0$, sehingga kembali ke PL mula - mula.

minimumkan : $\bar{c}\bar{x}$

Pembatas - pembatas : $A\bar{x} = \bar{b}$

$$\bar{x} \geq 0$$

Selesaikan persoalan ini terlebih dahulu, sehingga akan ada 3 kemungkinan masalah yang dihadapi :

masalah 1 : PL-nya tidak fisibel.

Dalam masalah ini PL parametriknya juga tidak fisibel untuk semua nilai dari s , sebab s berubah sebesar apapun, pembatas - pembatas pada PL parametriknya tetap tidak berubah.

masalah 2 : PL-nya unbounded.

Dalam masalah ini Algoritma Simplek telah menghasilkan SFB \bar{x} , dan juga \bar{d} yang menyebabkan unbounded ($\bar{d} \geq 0$).

Sesuai dengan perubahan s dari 0 ke suatu nilai, maka koefisien fungsi tujuannya juga mengalami perubahan dari \bar{c} ke $\bar{c} + s\bar{c}^1$.

Untuk menghindari unboundedness (jika memungkinkan), pilih s sedemikian sehingga $(\bar{c} + s\bar{c}^1)_N - (\bar{c} + s\bar{c}^1)_B B^{-1}N \geq 0$; $s \geq 0$.

Kadang - kadang unboundedness tidak dapat dihindari, karena tidak ada s yang memenuhi $(\bar{c} + s\bar{c}^1)_N - (\bar{c} + s\bar{c}^1)_B B^{-1}N \geq 0$.

masalah 3 : Sebuah solusi optimal \bar{x}^* dari PL mula - mula telah diperoleh $\bar{x}^* = (\bar{x}_B^*, \bar{x}_N^*) = (B^{-1}\bar{b}, 0) \geq 0$.

Langkah selanjutnya adalah menentukan solusi yang memenuhi interval - interval tertentu.

Langkah 1 : Menentukan nilai s agar \bar{x}^* tetap optimal.

Diakhir langkah 0 telah diperoleh \bar{x}^* yang memenuhi $\bar{c}_N - \bar{c}_B B^{-1}N \geq 0$.

Sesuai dengan perubahan s dari 0 ke suatu nilai, maka koefisien fungsi tujuan berubah dari \bar{c} ke $\bar{c} + s\bar{c}^1$.

Demikian juga dengan \bar{c}_N berubah menjadi $(\bar{c} + s\bar{c}^1)_N$ dan \bar{c}_B berubah menjadi $(\bar{c} + s\bar{c}^1)_B$.

Oleh sebab itu harga reduksinya berubah dari $\bar{c}_N - \bar{c}_B B^{-1}N$ ke $(\bar{c}_N + s\bar{c}_N^1) - (\bar{c}_B + s\bar{c}_B^1)B^{-1}N$.

Ambil $Y = \bar{c}_N - \bar{c}_B B^{-1}N$ dan $Y^1 = \bar{c}_N^1 - \bar{c}_B^1 B^{-1}N$, sejauh s yang dipilih itu memenuhi $Y + sY^1 \geq 0$, maka SFB \bar{x}^* tetap optimal.

Pilih s^* sebagai nilai terbesar dari s yang memenuhi $Y + sY^1 \geq 0$.

Karena PL mula - mula sudah mencapai optimum, maka pastilah $Y = \bar{c}_N - \bar{c}_B B^{-1}N \geq 0$, sehingga untuk menghitung s^* yang perlu diperhatikan adalah :

1. Jika $Y^1 \geq 0$, maka $s^* = \infty$

Karena $\forall s \geq 0$ dan $Y + sY^1 \geq 0$, maka s itu dapat diganti dengan sembarang angka.

2. Jika $Y^1 < 0$

maka $s^* = \min \frac{-Y_j}{Y_j^1} ; 1 \leq j \leq n-m$ dan $Y_j^1 < 0 ; s^* = \frac{-Y_{j^*}}{Y_{j^*}^1}$

dipilih s^* yang minimum dari $\frac{-Y_j}{Y_j^1}$, karena agar semua j memenuhi

$$(Y_j + sY_j^1) \geq 0$$

$$\begin{matrix} >0 & <0 \end{matrix}$$

Sehingga diperoleh :

$\forall s ; 0 \leq s \leq s^*$, SFB x^* adalah tetap optimal

Langkah selanjutnya menggambarkan bagaimana caranya untuk

melanjutkan ke interval berikutnya dan menemukan matrik basis

This document is the property of Universitas Diponegoro (UNDIP-IR) and may, without changing the content, translate the submission to any medium or format for the purpose of preservation. The author(s) or copyright owner(s) yang baru. UNDIP-IR may keep more than one copy of this submission for purpose of security, back-up and preservation.

(<http://eprints.undip.ac.id>)

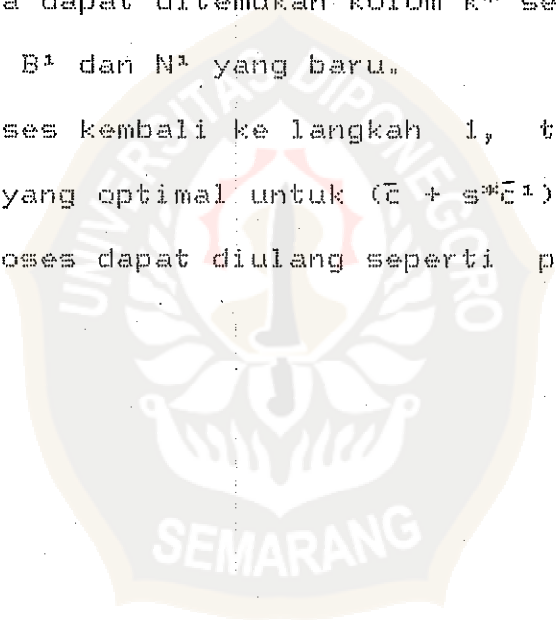
Langkah 2 : Menemukan sfb baru.

Untuk semua nilai $s > s^*$, komponen ke j^* harga reduksi dari $\bar{c} + s\bar{c}^1$ akan < 0 , oleh sebab itu dicari SFB yang baru dengan memasukkan N_{j^*} sebagai basis, dan kemudian dihitung $\bar{d} = (-B^{-1}N_{j^*}, I_{j^*})$ seperti biasanya.

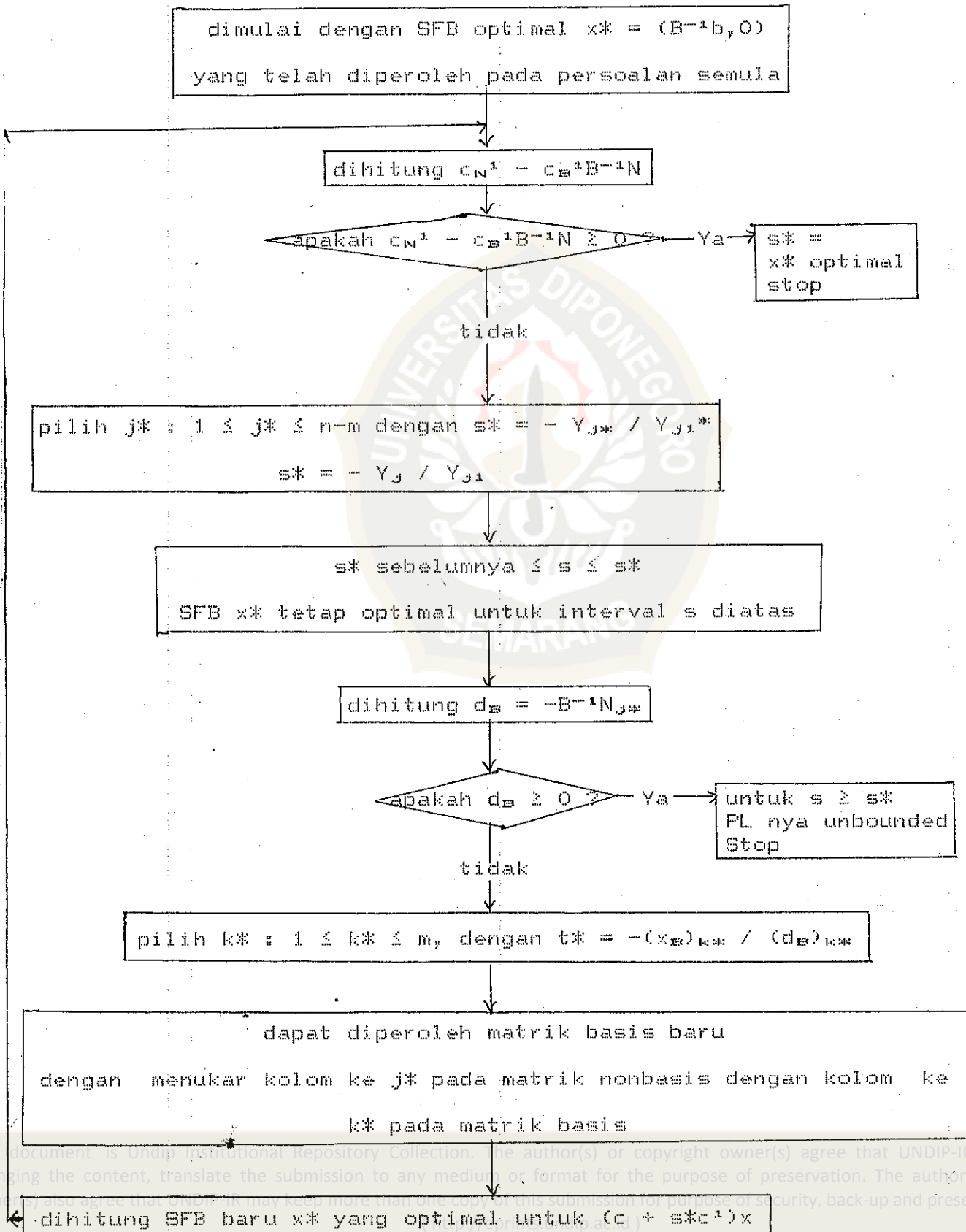
- Jika $\bar{d}_k \geq 0$; $\forall s \geq s^*$, PL parametriknya menjadi unbounded.
- Jika $\bar{d}_k < 0$; maka dapat ditemukan kolom k^* seperti biasanya, dan dapat diperoleh B^1 dan N^1 yang baru.

Sebelum seluruh proses kembali ke langkah 1, terlebih dahulu dihitung SFB baru \bar{x}^* yang optimal untuk $(\bar{c} + s^*\bar{c}^1)\bar{x}$

Kemudian seluruh proses dapat diulang seperti pada langkah 1 lagi.



FLOW CHART PARAMETRIK FUNGSI TUJUAN



PROGRAM PARAMETRIK FUNGSI TUJUAN

```
100 DIM FT(9),FTS(9),FP(9,9),B(9),X$(9),B1(9,9),BB(9,9),P(9,9),
    Q(9,9),NN(9,9),XX(9),BI(9),NI(9),C1(9,9),CN(9),CB(9),DB(9),
    DN(9),D(9),MIN#(9)
110 DIM CNS(9),CBS(9),DBS(9),DNS(9),DS(9),CBN(9)
120 MIN#(7)=1234567!:MIN#(8)=12345678#:MIN#(9)=123456789#
130 MIN#(2)=12:MIN#(3)=123:MIN#(4)=1234:MIN#(5)=12345!:
    MIN#(6)=123456!
140 X$(1)="X1":X$(2)="X2":X$(3)="X3":X$(4)="X4":X$(5)="X5"
150 X$(6)="X6":X$(7)="X7":X$(8)="X8":X$(9)="X9"
160 CLS
170 LOCATE 1,1:INPUT "Jumlah 'X-i' dari fungsi tujuan C <1 .
    . 9> : ",A1
180 IF A1<1 OR A1> 9 THEN BEEP:LOCATE 1,1:PRINT SPC(79):GOTO 170
190 LOCATE 2,1:INPUT "Jumlah 'X-i' dari fungsi tujuan C' <1 .
    . 9> : ",A4
200 IF A4<1 OR A4> 9 THEN BEEP:LOCATE 2,1:PRINT SPC(79):GOTO 190
210 LOCATE 3,1:INPUT "Jumlah 'X-i' dari fungsi pembatas <1 .
    . 9> : ",A2
220 IF A2<1 OR A2> 9 THEN BEEP:LOCATE 3,1:PRINT SPC(79):GOTO 210
230 LOCATE 4,1:INPUT "Jumlah persamaan fungsi pembatas <1 .
    . 9> : ",A3
240 IF A3<1 OR A3> 9 THEN BEEP:LOCATE 4,1:PRINT SPC(79):GOTO 230
250 PRINT:PRINT"Fungsi Tujuan C <masukan X-i kemudian tekan ente
    r> : "
260 FOR I = 1 TO A1:LOCATE 8,5+(I-1)*7:PRINT X$(I):NEXT I
270 PRINT:PRINT"Fungsi Tujuan C' <masukan X-i kemudian tekan ente
    r> : "
280 FOR I = 1 TO A4:LOCATE 12,5+(I-1)*7:PRINT X$(I):NEXT I
290 LOCATE 14,1:PRINT"Fungsi Pembatas <masukan X-i kemudian teka
    n enter> : "
300 FOR J = 1 TO A3:FOR I = 1 TO A2
310 LOCATE 15+J,4+(I-1)*6:PRINT X$(I)
320 NEXT I:LOCATE 15+J,7+(I-2)*6:PRINT"=":NEXT J
330 LOCATE 16+J,5:FOR I = 1 TO A2-1:PRINT X$(I);", ";:NEXT I:PRIN
    T X$(A2);" >= 0"
340 FOR I = 1 TO A1:LOCATE 8,1+(I-1)*7:INPUT"",FT(I):NEXT I
350 FOR I = 1 TO A4:LOCATE 12,1+(I-1)*7:INPUT"",FTS(I):NEXT I
360 FOR J = 1 TO A3:FOR I = 1 TO A2
370 LOCATE 15+J,1+(I-1)*6:INPUT"",FP(J,I)
380 NEXT I:LOCATE 15+J,9+(I-2)*6:INPUT"",B(J):NEXT J
390 PRINT:PRINT:PRINT
400 LPRINT"Fungsi Tujuan C : "
410 LPRINT FT(1);X$(1);" ";
420 FOR I = 2 TO A1
430 IF FT(I) < 0 THEN LPRINT FT(I);X$(I);" ";:GOTO 450
440 LPRINT"+";FT(I);X$(I);" ";
450 NEXT I:LPRINT
460 LPRINT:LPRINT"Fungsi Tujuan C' : "
470 LPRINT FTS(1);X$(1);" ";
480 FOR I = 2 TO A4
490 IF FTS(I) < 0 THEN LPRINT FTS(I);X$(I);" ";:GOTO 510
500 LPRINT"+";FTS(I);X$(I);" ";
```

```

510 NEXT I:LPRINT
520 LPRINT:LPRINT"Fungsi Pembatas : "
530 FOR J = 1 TO A3:LPRINT FP(J,1);X$(1);" ";
540 FOR I = 2 TO A2
550 IF FP(J,I) < 0 THEN LPRINT FP(J,I);X$(I);" ";GOTO 570
560 LPRINT"+";FP(J,I);X$(I);" ";
570 NEXT I:LPRINT"=";B(J):NEXT J:LPRINT
580 FOR I = 1 TO A2-1:LPRINT X$(I);", ";:NEXT I:LPRINT X$(A2);
  CHR$(242);"0":LPRINT:LPRINT:SW=0:ITER=0
590 Z$=""
600 FOR I = 1 TO A3:Z$=Z$+STR$(A2+1-I):NEXT I:U#=VAL(Z$)
610 GOSUB 2130
620 FOR I = 1 TO A3:BI(I)=VAL(MID$(U$,I,1)):NEXT I
630 H=0:FOR I = 1 TO A2:FOR J = 1 TO A3
640 IF I = BI(J) THEN 660
650 NEXT J:H=H+1:NI(H)=I
660 NEXT I
670 FOR J = 1 TO A3:FOR I = 1 TO A3:BI(J,I)=FP(J,BI(I)):NEXT I:NE
  XT J
680 ITER=ITER+1:LPRINT:LPRINT"ITERASI : ";ITER:LPRINT
690 ' ***** LANGKAH KE-0 *****
700 LPRINT"Basisnya = ";:FOR I = 1 TO A3-1:LPRINT X$(BI(I));" , "
  ;:NEXT I:LPRINT X$(BI(I)):LPRINT
710 LPRINT"Nonbasis = ";:FOR I = 1 TO A2-A3-1:LPRINT X$(NI(I));"
  , ";:NEXT I:LPRINT X$(NI(I)):LPRINT
720 LPRINT"Matrik Basis :";:FOR I = 1 TO A3:FOR J = 1 TO A3:LPRIN
  T BI(I,J);:NEXT J:LPRINT:LPRINT" ";:NEXT I:LPRIN
  T
730 GOSUB 1690
740 LPRINT"Inverse Matrik Basis :";:FOR I = 1 TO A3:FOR J = 1 TO
  A3:LPRINT USING "###.###" ";BB(I,J);:NEXT J:LPRINT:LPRINT"
  ";:NEXT I:LPRINT
750 FOR J = 1 TO A3:FOR I = 1 TO A1:NN(J,I)=FP(J,NI(I)):NEXT I:NE
  XT J
760 LPRINT"Matrik Nonbasis :";:FOR I = 1 TO A3:FOR J = 1 TO A2-
  A3:LPRINT NN(I,J);:NEXT J:LPRINT:LPRINT" ";
  :NEXT I:LPRINT
770 LPRINT"Xb : ";
780 FOR I = 1 TO A2-A3:XX(NI(I))=0:NEXT I
790 FOR J = 1 TO A3:Z=0
800 FOR I = 1 TO A3:Z=Z+BB(J,I)*B(I):NEXT I
810 IF Z < 0 THEN LPRINT"*** GANTI BASIS ***":GOTO 610
820 XX(BI(J))=Z:LPRINT X$(BI(J));" = ";USING "###.###" ";Z;:NE
  XT J:LPRINT:LPRINT
830 LPRINT"X :";:FOR I = 1 TO A2:LPRINT USING "###.###" ";XX(I
  );:NEXT I:LPRINT
840 ' ***** LANGKAH KE-1 *****
850 FOR K = 1 TO A2-A3
860 FOR J = 1 TO A3:Z=0
870 FOR I = 1 TO A3
880 Z=Z+BB(J,I)*NN(I,K):NEXT I
890 C1(J,K)=Z:NEXT J
900 NEXT K
910 IF SW=0 THEN 1030
920 FOR I = 1 TO A1:CNS(I)=FTS(NI(I)):NEXT I
930 FOR I = 1 TO A3:CBS(I)=FTS(BI(I)):NEXT I

```



```

940 FOR K = 1 TO A2-A3:Z=0
950 FOR I = 1 TO A3
960 Z=Z+CBS(I)*C1(I,K):NEXT I
970 CNS(K)=CNS(K)-Z:NEXT K
980 LPRINT:LPRINT"Y' :";:FOR I = 1 TO A2-A3:LPRINT CNS(I);:NEXT I:
LPRINT:LPRINT
990 FOR I = 1 TO A2-A3: IF CNS(I) < 0 THEN 1030
1000 NEXT I
1010 LPRINT:LPRINT"*** Jadi untuk X = (";:FOR I = 1 TO A2:LPRINT
XX(I);",":NEXT I:LPRINT")"
1020 LPRINT"adalah optimal untuk semua s, dimana : s >= ";LPRINT
T USING"###.###";BTS(DD):LPRINT:LPRINT:END
1030 IF SW=1 THEN FOR I = 1 TO A2:FT(I)=FT(I)+S*FTS(I):NEXT I
1040 IF SW=1 THEN LPRINT"S: ";USING"###.###";S
1050 FOR I = 1 TO A2-A3:CN(I)=FT(NI(I)):NEXT I
1060 FOR I = 1 TO A3:CB(I)=FT(BI(I)):NEXT I
1070 FOR K = 1 TO A2-A3:Z=0
1080 FOR I = 1 TO A3
1090 Z=Z+CB(I)*C1(I,K):NEXT I
1100 CN(K)=CN(K)-Z:NEXT K
1110 LPRINT:LPRINT"Y :";:FOR I = 1 TO A2-A3:LPRINT CN(I);:NEXT I:LPR
INT:LPRINT
1120 FOR I = 1 TO A2-A3: IF CN(I) < 0 THEN 1210
1130 NEXT I
1140 IF SW=1 THEN 1250
1150 LPRINT"*****"
1160 LPRINT"*** X adalah SFB yang optimal ***"
1170 LPRINT"*****"
1180 LPRINT
1190 SW=1:DD=0:BTS(DD)=0
1200 FOR I = 1 TO A1:CBN(I)=FT(I):NEXT I:FOR J = I TO 9:CBN(J)=0:
NEXT J:GOTO 670
1210 PTRJ=1:W=CN(1):FOR I = 2 TO A1: IF W > CN(I) THEN W=CN(I):PT
RJ=I
1220 NEXT I
1230 LPRINT"Yang terkecil (j*) adalah :";PTRJ:LPRINT
1240 IF SW=0 THEN 1390
1250 FOR I = 1 TO 9:P(1,I)=0:P(2,I)=0:NEXT I
1260 Y=0:FOR I = 1 TO A2-A3
1270 IF CNS(I)<0 THEN Y=Y+1:P(1,Y)=I
1280 NEXT I
1290 FOR I = 1 TO Y:S=-(CN(P(1,I))/CNS(P(1,I))):P(2,I)=S:NEXT I
1300 PTRJS=P(1,1):W=P(2,1):FOR I = 2 TO Y:IF W > P(2,I) THEN W=P(
2,I):PTRJS=P(1,I)
1310 NEXT I
1320 S=W
1330 DD=DD+1:BTS(DD)=S:IF DD > 1 THEN BTS(DD)=BTS(DD)+BTS(DD-1)
1340 LPRINT"Yang terkecil (s*) adalah :";PTRJS;" dengan harga s
:";USING"###.###";S:LPRINT
1350 IF DD=1 THEN LPRINT:LPRINT"*** Jadi untuk X = (";:FOR I = 1
TO A2:LPRINT XX(I);",":NEXT I:LPRINT")"
1360 IF DD=1 THEN LPRINT"adalah optimal untuk semua s, dimana :
0 <= s <= ";LPRINT USING"###.###";BTS(DD):LPRINT:LPRINT:GOT
O 1390
1370 LPRINT:LPRINT"*** Jadi untuk X = (";:FOR I = 1 TO A2:LPRINT

```

```

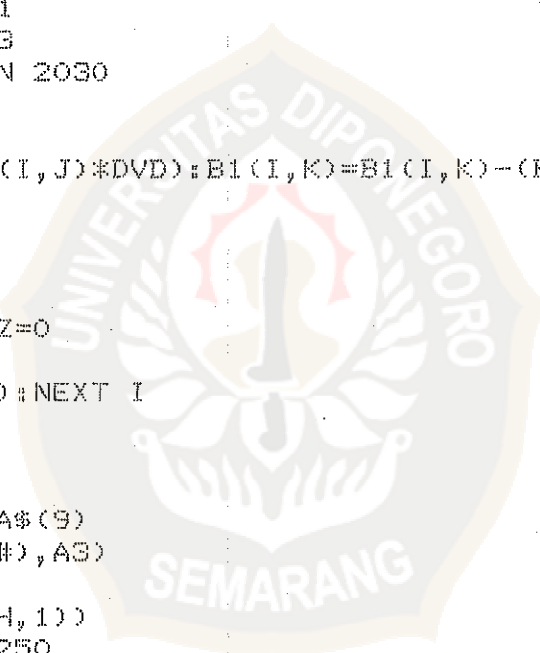
XX(I);",",":NEXT I:LPRINT")"
1380 LPRINT"adalah optimal untuk semua s, dimana : ";LPRINT USIN
G "###.###";BTS(DD-1);:LPRINT" =< s =< ";LPRINT USING"###.#
##";BTS(DD):LPRINT:LPRINT
1390 ' ***** LANGKAH KE-2 *****
1400 LPRINT"Db:";:IF SW=1 THEN PTRJ=PTRJS
1410 FOR J = 1 TO A3:Z=0
1420 FOR I = 1 TO A3:Z=Z-BB(J,I)*NN(I,PTRJ):NEXT I
1430 DB(J)=Z:LPRINT USING"###.###" ";DB(J);:NEXT J:LPRINT:LPRIN
T
1440 FOR I = 1 TO A1:DN(I)=0:NEXT I:DN(PTRJ)=1
1450 FOR I = 1 TO A1:D(NI(I))=DN(I):NEXT I
1460 FOR I = 1 TO A3:D(BI(I))=DB(I):NEXT I
1470 LPRINT"D :";:FOR I = 1 TO A2:LPRINT USING"###.###" ";D(I)
;:NEXT I:LPRINT:LPRINT
1480 ' ***** LANGKAH KE-3 *****
1490 Z=0:FOR I = 1 TO A3:IF DB(I)<0 THEN Z=Z+1
1500 NEXT I
1510 IF Z=0 THEN LPRINT "***** PL-nya UNBOUNDED *****":END
1520 IF Z=1 THEN 1610
1530 FOR I = 1 TO Z:P(1,I)=0:P(2,I)=0:NEXT I
1540 Y=0:FOR I = 1 TO A3
1550 IF DB(I)<0 THEN Y=Y+1:P(1,Y)=I
1560 NEXT I
1570 FOR I = 1 TO Y:T=-(XX(BI(P(1,I)))/DB(P(1,I))):P(2,I)=T:NEXT
I
1580 PTRK=P(1,1):W=P(2,1):FOR I = 2 TO Y:IF W > P(2,I) THEN W=P(2
,I):PTRK=P(1,I)
1590 NEXT I
1600 T=W:GOTO 1640
1610 FOR I = 1 TO A3:IF DB(I)<0 THEN PTRK=I
1620 NEXT I
1630 T=-XX(BI(PTRK))/DB(PTRK)
1640 LPRINT"T =";T;" k* =";PTRK:LPRINT
1650 ' ***** LANGKAH KE-4 *****
1660 W=BI(PTRK):BI(PTRK)=NI(PTRJ):NI(PTRJ)=W
1670 GOTO 670
1680 END
1690 FOR I = 1 TO A3:FOR J = 1 TO A3:P(I,J)=0:Q(I,J)=0:NEXT J:NEX
T I
1700 FOR I = 1 TO A3:P(I,I)=1:Q(I,I)=1:NEXT I
1710 ' ***** OPERASI BARIS *****
1720 FOR K = 1 TO A3
1730 IF B1(K,K) = 1 THEN 1870
1740 IF B1(K,K) <> 0 THEN 1830
1750 FOR L = 1 TO A3
1760 IF B1(L,K) = 0 THEN 1810
1770 FOR M = 1 TO A3
1780 P(K,M)=P(K,M)+P(L,M):B1(K,M)=B1(K,M)+B1(L,M)
1790 NEXT M
1800 L=A3
1810 NEXT L
1820 GOTO 1730
1830 DVD = B1(K,K)
1840 FOR I = 1 TO A3

```

```

1850 P(K, I)=P(K, I)/DVD: B1(K, I)=B1(K, I)/DVD
1860 NEXT I
1870 FOR J = K+1 TO A3
1880 IF B1(J, K)=0 THEN 1930
1890 DVD = B1(J, K)
1900 FOR I = 1 TO A3
1910 P(J, I)=P(J, I)-(P(K, I)*DVD): B1(J, I)=B1(J, I)-(B1(K, I)*DVD)
1920 NEXT I
1930 NEXT J
1940 NEXT K
1950 * ***** OPERASI KOLOM *****
1960 FOR J = 1 TO A3-1
1970 FOR K = J+1 TO A3
1980 IF B1(J, K)=0 THEN 2030
1990 DVD = B1(J, K)
2000 FOR I = 1 TO A3
2010 Q(I, K)=Q(I, K)-(Q(I, J)*DVD): B1(I, K)=B1(I, K)-(B1(I, J)*DVD)
2020 NEXT I
2030 NEXT K
2040 NEXT J
2050 FOR K = 1 TO A3
2060 FOR J = 1 TO A3: Z=0
2070 FOR I = 1 TO A3
2080 Z=Z+Q(J, I)*P(I, K): NEXT I
2090 BB(J, K)=Z: NEXT J
2100 NEXT K
2110 RETURN
2120 DIM MIN$(9), ANGKA$(9)
2130 U$=RIGHT$(STR$(U#), A3)
2140 FOR H = 1 TO A3
2150 V = VAL(MID$(U$, H, 1))
2160 IF V > A2 THEN 2250
2170 IF V = 0 THEN 2250
2180 FOR HH = 1 TO A3
2190 W = VAL(MID$(U$, HH, 1))
2200 IF V=W AND H>HH THEN 2250
2210 NEXT HH
2220 NEXT H
2230 U# = U# - 1
2240 RETURN
2250 U# = U# - 1: U$=STR$(U#): U$=RIGHT$(U$, A3)
2260 IF U# < MIN$(A3) THEN PRINT"***** PL-nya TIDAK FISIBEL *****"
    ":END
2270 GOTO 2140

```



PENJELASAN PROGRAM PARAMETRIK FUNGSI TUJUAN

- 100 Dimensi / array, yaitu suatu pemesanan suatu tempat yang diwakili oleh suatu variabel tertentu, yaitu :
- FT(9) disini disediakan untuk fungsi tujuan dengan maksimum 9 buah X_1 .
- FTS(9) disini disediakan untuk fungsi tujuan tambahan apabila terjadi penambahan fungsi tujuan dengan rumus $C=C+SC^1$, maksimum X_1 -nya 9 buah.
- FP(9,9) disediakan untuk fungsi pembatas dengan maksimum 9 buah X_1 dan 9 buah persamaan sebagai fungsi pembatas.
- B(9) disediakan untuk harga b dari setiap persamaan yang ada sehingga maksimum b-nya adalah 9 buah.
- B1(9,9) disediakan untuk matriks basisnya, ukuran matriks basis maksimum adalah 9×9 .
- BB(9,9) disediakan untuk inverse matriks basis.
- P(9,9) dan Q(9,9) disediakan untuk perhitungan pencarian inverse matriks basis dengan menggunakan transformasi elementer.
- NN(9,9) disediakan untuk matriks nonbasis.
- XX(9) disediakan untuk hasil operasi X
- BI(9) dan NI(9) disediakan untuk urutan X_1 dari matriks basis dan matriks nonbasis.
- C1(9) disediakan untuk hasil perhitungan $B^{-1}N$.
- CB(9) disediakan untuk C_B .
- CN(9) disediakan untuk C^N .
- DB(9) disediakan untuk $-B^{-1}N_{j,k}$
- DN(9) disediakan untuk d_N
- D(9) disediakan untuk d
- MIN#(9) disediakan untuk kombinasi dari basis yang terkecil.
- 110 CNS(9) disediakan untuk C^1_N
- CBS(9) disediakan untuk C^1_B
- DBS(9) disediakan untuk d_B
- DNS(9) disediakan untuk d_N
- DS(9) disediakan untuk d.
- CBN(9) disediakan untuk C.
- 120 dan 130 digunakan untuk memberi harga minimal dalam kombinasi matriks basis, disini diperlukan karena pemilihan basis ditentukan secara acak oleh komputer dan dicoba untuk setiap kemungkinan sampai LP-nya adalah infisibel.
- 140 dan 150 X*(9) digunakan untuk mencetak dilayar / monitor harga X_1 .
- 160 Clear Screen dapat diartikan sebagai penghapus layar monitor dan posisi kursor berada pada lokasi 1,1 (sudut kiri atas)
- 170 sampai 240 memasukan jumlah X_1 dari fungsi tujuan dan fungsi pembatas dan jumlah persamaan pada fungsi pembatas.
- 250 sampai 380 proses pemasukan data.
- 590 dan 600 digunakan untuk mencari kombinasi terbesar dari kemungkinan kombinasi basis mxm sesuai dengan jumlah persamaan dan jumlah X_1 dari fungsi pembatas.
- 610 gosub 2130, line 2130 sampai dengan 2270 dilakukan pencarian kombinasi basis yang benar.
- 620 U\$ adalah variabel alfanumerik yang berisi kombinasi basis yang benar, tiap digit dari U\$ disimpan dalam array BI agar mempermudah pengoperasiannya.

630 sampai 660 dilakukan penyimpanan X_i nonbasis dalam array NI.
 390 sampai 580 dilakukan pencetakan ke kertas dari fungsi tujuan C, fungsi tujuan C^1 , fungsi pembatas dan pembatas non negatif.
 670 dilakukan pengisian matriks B1, yang merupakan matriks basis dari fungsi pembatas.

690 sampai 830 merupakan step ke-0.
 700 pencetakan harga B, basis yang dipakai.
 710 pencetakan harga N, nonbasis yang dipakai.
 720 pencetakan matriks basis.
 730 gosub 1690, pencarian inverse matriks dengan menggunakan transformasi elementer.
 740 pencetakan inverse matriks basis.
 750 dilakukan pengisian matriks NN yang merupakan matriks nonbasis dari fungsi pembatas.
 760 pencetakan matriks nonbasis.
 770 pencetakan X_B .
 780 isi array XX yang ke NI dijadikan 0 (X yang nonbasis selalu 0)
 800 perhitungan inverse matriks basis dikalikan dengan B, yang hasilnya disimpan ke Z.
 810 bila $Z < 0$ (berarti $X_B < 0$), maka pemilihan basis matriks perlu dirubah. Eksekusi selanjutnya ke 420
 820 nilai Z disimpan di array XX yang ke B1. $X_B = Z$.
 pencetakan nilai X_B .
 830 pencetakan nilai X.

840 sampai 1380 merupakan Langkah ke 1
 850 sampai 900 dilakukan perkalian inverse matriks basis dengan matriks nonbasis ($B^{-1} \times N$), yang hasilnya disimpan di matriks C1.
 920 sampai 1020 bila dilakukan perubahan C. $C=C+SC^1$.
 920 pengisian array CNS dengan elemen fungsi tujuan baru yang nonbasis.
 930 pengisian array CBS dengan elemen fungsi tujuan baru yang basisnya.
 960 perkalian isi array CBS dengan isi matriks C1, hasilnya disimpan di Z.
 C1 adalah $B^{-1} \times N$. CBS adalah C^1_B
 970 pengurangan isi array CNS dengan Z.
 CNS adalah C^1_N . Z adalah $C^1_B \times B^{-1} \times N$.
 Hasil perhitungan tsb disimpan di CNS. Jadi CNS berisi Y^1 .
 980 pencetakan nilai Y^1
 990 dan 1000 pengecekan nilai Y^1 apakah < 0 .
 Jika < 0 maka eksekusi dilanjutkan ke 810.
 Jika semua nilai Y^1 sudah > 0 , maka proses selesai.
 1010 dan 1020 pencetakan nilai S.
 1030 jika dilakukan perubahan atas C, dilakukan perhitungan C atau fungsi tujuannya, isi array FTS dikalikan dengan S dan dijumlahkan dengan FT dan hasilnya disimpan di FT.
 $C = C + S \times C^1$.
 1040 pencetakan nilai S, jika C berubah.
 1050 pengisian array CN dengan elemen fungsi tujuan yang ke NI, elemen yang nonbasis.

1060 pengisian array CB dengan elemen fungsi tujuan yang ke BI, elemen basis.

1090 perkalian isi array CB dengan isi matriks C1, hasilnya disimpan di Z.
C1 adalah $B^{-1} \times N$. CB adalah C_B

1100 pengurangan isi array CN dengan Z.
CN adalah C_N . Z adalah $C_B \times B^{-1} \times N$.
Hasil perhitungan tsb disimpan di CN. Jadi CN berisi Y.

1110 pencetakan nilai Y.

1120 pengecekan apakah nilai Y ada yang < 0 , bila ada eksekusi dilanjutkan ke 970.

1140 jika C berubah, eksekusi ke 1210

1150 sampai 1200 proses pengolahan dengan Algoritma Simplek selesai, eksekusi dilanjutkan ke 670

1210 sampai 1220 dicari nilai terkecil dari Y, yang akan digunakan sebagai j^* .

1230 pencetakan nilai j^* .

1240 Jika fungsi tujuan belum ditambah C^1 , eksekusi dilanjutkan ke 1390

1260 sampai 1280 dilakukan pencarian CNS yang < 0

1290 dilakukan perhitungan s

1300 mencari harga s yang paling kecil

1330 mencari interval dari s

1340 pencetakan harga s dan s^*

1350 sampai 1380 mencari interval dari s

1390 sampai 1470 langkah ke 2

1400 sampai 1430 mencari nilai dB dan pencetakan nilai dB

1440 mencari nilai dN

1450 dan 1460 mencari nilai d

1470 mencetak nilai d

1480 sampai 1640 langkah ke 3

1490 dan 1500 mencari banyaknya nilai dari dB yang < 0

1510 jika tidak ada nilai yang < 0 maka PLnya Unbounded

1520 Jika ada sebuah nilai dari dB yang < 0 , eksekusi dilanjutkan ke 1610

1530 sampai dengan 1600 Jika ada beberapa nilai dari d_B yang < 0 , maka dicari nilai yang paling kecil dari $-X_B/d_B$ dan nilai yang diperoleh adalah T, kemudian eksekusi dilanjutkan ke 1640

1610 dan 1620 mencari urutan dari D_B yang < 0

1630 dan 1640 mencari nilai dari T

1640 mencetak nilai T dan urutannya k^*

1650 sampai 1670 langkah ke 4

1660 diperoleh matrik Basis dan Nonbasis yang baru, dengan jalan menukar kolom ke j^* pada matrik Nonbasis dengan kolom ke k^* pada matrik Basis

1670 kemudian dilanjutkan pencarian nilai X SFB seperti diatas sehingga diperoleh nilai yang optimum, eksekusi dimulai dari 670

1690 dan 1700 merubah isi matriks P dan Q sehingga menjadi matriks identitas dengan ukuran $b \times b$.

1710 sampai 1940 dilakukan operasi baris dengan menggunakan matriks P.

1720 sampai 1940 dilakukan looping sebanyak b kali.

1730 pengecekan apakah nilai matriks B1 yang ke K,K (1,1 atau 2,2 ... b,b) apakah sudah = 1, bila sudah eksekusi dilanjutkan ke 1870

1740 pengecekan apakah nilai matriks B1 yang ke K,K (1,1 atau 2,2 ... b,b) apakah = 0, bila tidak eksekusi dilanjutkan ke 1830

1750 sampai 1810 karena nilai matriks B1 yang ke K,K masih = 0, maka perlu dijadikan $\neq 0$ dengan salah satu baris dari matriks B1.

1760 sampai 1810 dicari isi baris matriks B1 yang kolom ke K-nya tidak = 0.

1770 sampai 1790 dilakukan penjumlahan dari baris ke K dengan baris ke L dari matriks B1 dan P, sehingga isi yang ke K,K dari matriks B1 $\neq 0$.

1800 looping 1750 sampai 1810 diakhiri.

1820 eksekusi dilanjutkan ke 1730.

1830 elemen matriks B1 yang ke K,K dijadikan pembagi dan disimpan kedalam variabel DVD, hal ini dimaksudkan agar elemen matriks yang ke K,K menjadi = 1.

1850 dilakukan looping sebanyak b (A3) kali, elemen matriks B1 dan P baris yang ke K dibagi dengan DVD.

1870 sampai 1930 dilakukan looping sebanyak b (A3) - K kali.

1880 jika elemen yang ke J,K sudah = 0, eksekusi dilanjutkan ke 1930.

1890 elemen matriks B1 yang ke J,K dijadikan pembagi dan disimpan kedalam variabel DVD.

1910 dilakukan looping sebanyak b (A3) kali, elemen matriks B1 dan P baris yang ke J dikurangi dengan elemen matriks yang ke K,I yang dibagi dengan DVD, sehingga elemen matriks B1 yang ke J,K akan menjadi 0.

1960 sampai 2040 dilakukan looping sebanyak b (A3) - 1 kali, dilakukan operasi kolom dengan menggunakan matriks Q.

1970 sampai 2030 dilakukan looping sebanyak b (A3) - J kali dengan tujuan untuk membuat 0 elemen matriks B1 yang ke J,K.

1980 jika elemen matriks B1 yang ke J,K sudah = 0, eksekusi dilanjutkan ke 2030.

1990 elemen matriks B1 yang ke J,K dijadikan pembagi dan disimpan dalam variabel DVD.

2010 dilakukan looping sebanyak b (A3) kali, elemen matriks B1 dan Q kolom yang ke J dikurangi dengan elemen matriks yang ke K,I yang dibagi dengan DVD, sehingga elemen matriks B1 yang ke J,K akan menjadi 0.

2050 sampai 2100 dilakukan perkalian matriks Q dan P, yang hasilnya disimpan dalam matriks BB. Matriks BB ini merupakan inverse matriks dari B1.

2110 kembali ke line yang memanggilnya.

2130 U# adalah variabel double precision, dimana untuk setiap variabel numerik / angka selalu diawali dengan 1 spasi, hal ini digunakan untuk tanda - dari bilangan itu, sehingga apabila variabel double precision dirubah ke dalam variabel alfanumerik, juga akan ada 1 spasi didepannya. Disini digunakan instruksi RIGHT\$ dengan tujuan untuk menghilangkan

1 spasi yang muncul, U\$ akan berisi kombinasi basis.

2140 sampai 2220 merupakan looping atau pengulangan sebanyak A3 kali, disini A3 adalah variabel yang berisi jumlah persamaan fungsi pembatas, pengulangan ini dilakukan dengan tujuan untuk menghasilkan kombinasi yang benar dari kombinasi basis yang akan digunakan dalam perhitungan selanjutnya.

2150 diambil 1 digit dari variabel U\$ disimpan dalam variabel V.

2160 di cek apakah digit tersebut lebih besar dari A2 (A2 berisi angka X_i yang paling besar), sehingga apabila variabel V lebih besar dari A2 berarti kombinasi itu bukan kombinasi yang benar dari kombinasi basis. Apabila V lebih besar dari A2, eksekusi selanjutnya ke 2010.

2170 di cek apakah variabel V adalah 0, karena 0 bukan merupakan bagian dari kombinasi yang benar dari kombinasi basis. Bila $V = 0$, eksekusi selanjutnya ke 2010.

2180 sampai 2210 merupakan looping atau pengulangan sebanyak A3 kali.

2190 diambil 1 digit dari variabel U\$ disimpan dalam variabel W.

2200 disini dilakukan perbandingan apakah $W = V$ dan $H <> HH$, apabila kondisi itu dipenuhi berarti kombinasi yang dihasilkan adalah angka kembar, yang mana kombinasi tersebut adalah bukan kombinasi yang benar. Apabila kondisi ini terpenuhi maka eksekusi selanjutnya akan ke 2010.

2230 U# dikurangi dengan 1, sehingga nilai U# berkurang. Hal ini dimaksudkan untuk mempersiapkan kombinasi berikutnya yang lebih kecil.

2240 kembali ke line yang memanggilnya.

2250 U# dikurangi 1, dan variabel double precision dijadikan variabel alfanumerik seperti pada line 2230.

2260 dilakukan pengetesan apakah U# lebih kecil dari kombinasi basis yang terkecil, bila kondisi diatas dipenuhi maka LP-nya infisibel.

2270 eksekusi dilanjutkan ke 2140.