



```

PROGRAM UTAMA;
USES CRT;
VAR   C,COST,SIGN           : ARRAY[1..51,1..51] OF INTEGER;
      RLABEL,CLABEL,IR     : ARRAY[1..51] OF INTEGER;
      I1,J1,MIN,MM,M,N,K,T,I,X,Y,
      J,NUMLINES,FLIP,SMALLEST,
      SIZE,I2,MAKMIN,ZVAL,KROW,M1,
      N2,J2,N1,S1,JCOL     : INTEGER;
      JAWAB,LAGI           : CHAR;JUDUL           : STRING[5];
      ULANG,KELUAR,YA_TIDAK : BOOLEAN;
LABEL STEP0,STEP1,STEP2,STEP3,STEP4,STEP5,STEP6,STEP7,STEP8;
{*****}

```

```

PROCEDURE GETINPUT;

```

```

VAR I,J : INTEGER;

```

```

BEGIN

```

```

  WRITE('MASUKKAN JUDUL : ');

```

```

  READLN(JUDUL);

```

```

  WRITE('MASUKKAN BANYAKNYA BARIS (M) : ');

```

```

  READLN(M);

```

```

  WRITE('MASUKKAN BANYAKNYA KOLOM (N) : ');

```

```

  READLN(N);

```

```

  WRITE('PILIH BENTUK MASALAH [1] MIN / [2] MAK : ');

```

```

  READLN(MAKMIN);

```

```

  WRITELN;

```

```

  IF M>N THEN

```

```

    FOR I:=1 TO M DO

```

```

      FOR J:= 1 TO M DO

```

```

        C[I,J]:=0

```

```

  ELSE

```

```

    FOR I:=1 TO N DO

```

```

      FOR J:=1 TO N DO

```

```

        C[I,J]:=0;

```

```

  (* PENGISIAN ONGKOS *)

```

```

  WRITELN('MASUKKAN ONGKOS : ');WRITELN;WRITE(' ');

```

```

  FOR I:=1 TO M DO

```

```

    FOR J:=1 TO N DO

```

```

      BEGIN

```

```

        WRITE('C[',I,',',',',J,','] = ');READ(C[I,J]);WRITE(' ');

```

```

      END;WRITELN;

```

```

  END;

```

```

PROCEDURE MORE;

```

```

BEGIN

```

```

  WRITE('TEKAN TOMBOL UNTUK MELANJUTKAN : ');

```

```

  JAWAB:=READKEY;

```

```

END;

```

```

PROCEDURE MATPRINT; (* MENCETAK MATRIX UKURAN M OLEH N *)

```

```

VAR I,J,N2,KCOL : INTEGER;V : STRING;

```

```

BEGIN

```

```

  FLIP:=-1;

```

```

  KROW:=0;

```

```

  WHILE FLIP=-1 DO

```

```

    BEGIN

```

```

KROW:=KROW+1;
I1:=(KROW-1)*12+1;
I2:=(KROW-1)*12+12;
IF I2>M THEN I2:=M;
M1:=I2-I1+1;
N2:=((N-1) DIV 7)+1;
FOR KCOL:=1 TO N2 DO
BEGIN
  J1:=(KCOL-1)*7+1;
  J2:=(KCOL-1)*7+7;
  IF J2>N THEN J2:=N;
  N1:=J2-J1+1;
  S1:=10+4*N1;
  IF (KROW>1) OR (KCOL>1) THEN CLRSCR;
  WRITELN;
  GOTOXY(S1,WHEREY);WRITELN('PEKERJAAN');
  WRITELN;
  FOR J:=J1 TO J2 DO
  BEGIN
    JCOL:=J-((J-1) DIV 7)*7;
    GOTOXY(11+JCOL*8,WHEREY);WRITE(J);
  END;WRITELN;
  FILLCHAR(V,14+8*N1,'-');
  V[0]:=CHAR(13+8*N1);
  WRITELN(V);
  FOR I:=I1 TO I2 DO
  BEGIN
    IF I=(M1 DIV 2)+1 THEN WRITE('PEKERJA');
    GOTOXY(11,WHEREY);WRITE(I);
    FOR J:=J1 TO J2 DO
    BEGIN
      JCOL:=J-((J-1) DIV 7)*7;
      GOTOXY(11+JCOL*8,WHEREY);WRITE(C[I,J]);
    END;WRITELN;
  END;
  WRITELN(V);
  IF(I2=M) AND (J2=N) THEN FLIP:=0;
  IF FLIP<>0 THEN MORE;
END;
END;MORE;WRITELN;
END;

```

```

PROCEDURE SETUP;
VAR LARGEST,I,J:INTEGER ;
BEGIN

```

```

  SIZE:=M;
  IF SIZE<N THEN SIZE:=N;
  FOR I:=1 TO SIZE DO
    FOR J:=1 TO SIZE DO
      COST[I,J]:=0;
  LARGEST:=-3000;
  FOR I:=1 TO M DO
    FOR J:=1 TO N DO
      BEGIN
        COST[I,J]:=C[I,J];

```

```

IF C[I,J]>LARGEST THEN LARGEST:=C[I,J];
END;
IF M<N THEN
BEGIN
WRITELN('BARIS KURANG DARI KOLOM,TAMBAHKAN BARIS SEMU');
WRITELN('BARIS SEMU AKAN DITAMBAH DENGAN NILAI TERBESAR/TERKECIL');

WRITELN('DALAM SETIAP SEL TERGANTUNG MASALAHNYA');
END
ELSE
IF M>N THEN
BEGIN
WRITELN('KOLOM KURANG DARI BARIS,TAMBAHKAN KOLOM SEMU');
WRITELN('KOLOM SEMU AKAN DITAMBAH DENGAN NILAI TERBESAR/TERKECIL');

WRITELN('DALAM SETIAP SEL TERGANTUNG MASALAHNYA');
END
ELSE
WRITELN('BARIS = KOLOM');
(* SEANDAINYA MASALAH MAK, KURANGKAN ELEMEN TERBESAR PADA *)
(* SEMUA ELEMEN DALAM MATRIX ONGKOS DAN MINIMALKAN *)
WRITE('PILIH MASALAH PENUGASAN [1] MIN/[2] MAK : ');
READLN(MAKMIN);
IF MAKMIN=2 THEN
BEGIN
FOR I:=1 TO SIZE DO
FOR J:=1 TO SIZE DO
COST[I,J]:=LARGEST-COST[I,J];
END;
END;
3END;

3BEGIN (* PROGRAM UTAMA *)
ULANG:=TRUE;
REPEAT
CLRSCR;
GETINPUT;
(* MULAI PROSES MENJAWAB *)
MATPRINT;
WRITELN;
SETUP;
(* MULAI PROSES MENJAWAB *)
WRITELN;
X:=WHEREX;Y:=WHEREY;
WRITE('JUDUL PROGRAM : MASALAH : ');
GOTOXY(19,Y);WRITE(JUDUL);
GOTOXY(45,Y);
IF MAKMIN=2 THEN WRITE('MAKSIMAL') ELSE WRITE('MINIMAL');
WRITELN;
WRITE('BANYAKNYA BARIS : BANYAKNYA KOLOM : ');
GOTOXY(19,Y);WRITE(M);
GOTOXY(45,Y);WRITE(N);
WRITELN;

```

```

FOR I:=1 TO SIZE DO
BEGIN
  J1:=1;
  FOR J:=2 TO SIZE DO
    IF COST[I,J]<COST[I,J1] THEN J1:=J;

    IF COST[I,J1]<>0 THEN
    BEGIN
      MIN:=COST[I,J1];
      FOR J:=1 TO SIZE DO COST[I,J]:=COST[I,J]-MIN;
    END;
  END;
* KURANGKAN ELEMEN TERKECIL DARI TIAP KOLOM PADA ELEMEN
  LAINNYA DALAM KOLOM ITU *)
  FOR J:=1 TO SIZE DO
  BEGIN
    I1:=1;
    FOR I:=2 TO SIZE DO
      IF COST[I,J]<COST[I1,J] THEN I1:=I;

      IF COST[I1,J]<>0 THEN
      BEGIN
        MIN:=COST[I1,J];
        FOR I:=1 TO SIZE DO COST[I,J]:=COST[I,J]-MIN;
      END;
    END;
  END;
(* MEMPERBESAR MATRIX *)
  COST[SIZE+1,SIZE+1]:=0;
  FOR I:=1 TO SIZE DO
  BEGIN
    COST[I,SIZE+1]:=0;
    RLABEL[I]:=0;
  END;
  FOR J:=1 TO SIZE DO
  BEGIN
    COST[SIZE+1,J]:=0;
    CLABEL[J]:=0;
  END;
  CLABEL[SIZE+1]:=1;
(* MENGADAKAN (PENANDAAN) MATRIX *)
  FOR I:=1 TO SIZE+1 DO
  FOR J:=1 TO SIZE+1 DO
  BEGIN
    IF COST[I,J]=0 THEN SIGN[I,J]:=0 ELSE SIGN[I,J]:=-1;
  END;

STEP1:
(* MENEMUKAN NOL YANG TIDAK DITANDAI DALAM BARIS YANG TAK DINAMAKAN
  DARI KOLOM I *)
  I:=1; KELUAR:=FALSE;
  REPEAT
  BEGIN
    IF RLABEL[I]=0 THEN
    IF SIGN[I,SIZE+1]=0 THEN
    BEGIN

```

```

        RLABEL[I]:=SIZE+1;
        KELUAR:=TRUE;
    END;
    INC(I);
END;
UNTIL (I>SIZE) OR KELUAR;

```

STEP2:

MENEMUKAN NOL YANG TIDAK DITANDAI DALAM BARIS YANG DINAMAKAN DAN KOLOM YANG TAK DINAMAKAN \*)

FOR I:=1 TO SIZE+1 DO

BEGIN

IF RLABEL[I] <> 0 THEN

FOR J:=1 TO SIZE+1 DO

BEGIN

IF CLABEL[J] = 0 THEN

IF SIGN[I,J]=0 THEN

BEGIN

CLABEL[J]:=I;

IF SIGN[SIZE+1,J]=0 THEN

BEGIN

RLABEL[SIZE+1]:=J;

GOTO STEP4;

END;

GOTO STEP3;

END;

END;

END;

STEP3:

(\* MENEMUKAN NOL YANG DITANDAI DALAM KOLOM YANG DINAMAKAN DAN BARIS YANG TAK DINAMAKAN \*)

FOR I:=1 TO SIZE DO

BEGIN

IF RLABEL[I]=0 THEN

FOR J:=1 TO SIZE DO

BEGIN

IF CLABEL[J] <> 0 THEN

IF SIGN[I,J] = +1 THEN

BEGIN

RLABEL[I]:=J;

GOTO STEP2;

END;

END;

END;

STEP4:

(\* SEANDAINYA BARIS D TAK DINAMAKAN MENUJU STEP7, SEBALIKNYA MASUKKAN TANDA NOL DALAM SEL (D,K) \*)

IF RLABEL[SIZE+1]=0 THEN GOTO STEP7;

K:=RLABEL[SIZE+1];

SIGN[SIZE+1,K]:=+1;

STEP5:

(\* MEMBUAT SISA NOL YANG BERBENTUK PATH \*)

```

T:=CLABEL[K];
SIGN[T,K]:=+1;
MM:=RLABEL[T];
IF MM=SIZE+1 THEN GOTO STEP6;
SIGN[T,MM]:=0;
K:=MM;
GOTO STEP5;

```

```

STEP6:
(* MASUKKAN TANDA POSITIF PADA SEL C(T,I), HAPUS NAMA DAN
MENCARI PATH BARU *)

```

```

SIGN[T,SIZE+1]:=+1;
FOR I:=1 TO SIZE DO
BEGIN

```

```

    RLABEL[I]:=0;
    CLABEL[I]:=0;

```

```

END;

```

```

CLABEL[SIZE+1]:=-1;

```

```

RLABEL[SIZE+1]:=0;

```

```

GOTO STEP1;

```

```

STEP7:

```

```

(* MENUTUP NOL DALAM BARIS YANG TAK DINAMAKAN DAN KOLOM
YANG DINAMAKAN *)

```

```

NUMLINES:=0;

```

```

FOR I:=1 TO SIZE DO

```

```

BEGIN

```

```

    IF RLABEL[I]=0 THEN NUMLINES:=NUMLINES+1;

```

```

    IF CLABEL[I]<>0 THEN NUMLINES:=NUMLINES+1;

```

```

END;

```

```

IF NUMLINES=SIZE THEN GOTO STEP8;

```

```

SMALLEST:=3000;

```

```

FOR I:=1 TO SIZE DO

```

```

BEGIN

```

```

    IF RLABEL[I] <> 0 THEN

```

```

        FOR J:=1 TO SIZE DO

```

```

            BEGIN

```

```

                IF CLABEL[J]=0 THEN

```

```

                    IF COST[I,J]<SMALLEST THEN SMALLEST:=COST[I,J];

```

```

            END;

```

```

        END;

```

```

(* PENGURANGAN TERKECIL DARI ELEMEN YANG TAK DITUTUPI *)

```

```

FOR I:=1 TO SIZE DO

```

```

BEGIN

```

```

    IF RLABEL[I] <> 0 THEN

```

```

        FOR J:=1 TO SIZE DO

```

```

            BEGIN

```

```

                IF CLABEL[J]=0 THEN COST[I,J]:=COST[I,J]-SMALLEST;

```

```

            END;

```

```

        END;

```

```

(* MENJUMLAHKAN TERKECIL PADA ELEMEN YANG DITUTUPI 2 GARIS *)

```

```

FOR I:=1 TO SIZE DO

```

```

BEGIN
  IF RLABEL[I] = 0 THEN
    BEGIN
      FOR J:=1 TO SIZE DO
        IF CLABEL[J] <> 0 THEN COST[I,J]:=COST[I,J]+SMALLEST;
      END;
    END;
  FOR I:=1 TO SIZE DO
    BEGIN
      RLABEL[I]:=0;
      CLABEL[I]:=0;
    END;
  CLABEL[SIZE+1]:=-1;
  RLABEL[SIZE+1]:=0;
  FOR I:=1 TO SIZE+1 DO
    FOR J:=1 TO SIZE+1 DO
      BEGIN
        IF COST[I,J]=0 THEN
          SIGN[I,J]:=0
        ELSE
          SIGN[I,J]:=-1;
        END;
      GOTO STEP1;

STEP8:
  (* NILAI OPTIMAL DITEMUKAN *)
  FOR I:=1 TO SIZE DO
    FOR J:=1 TO SIZE DO
      IF SIGN[I,J]=+1 THEN IR[I]:=J;

  ZVAL:=0;
  FOR I:=1 TO M DO
    ZVAL:=ZVAL+C[I,IR[I]];
  WRITELN('NILAI OPTIMAL : ',ZVAL);

  FOR I:=1 TO SIZE DO
    BEGIN
      IF IR[I] <= N THEN
        BEGIN
          IF I-(I DIV 15)*15 = 0 THEN MORE;
          WRITE('PEKERJA : ',I);
          WRITELN(' DITUGASKAN KE PEKERJAAN : ',IR[I]);
        END;
      END;
    WRITELN;
    WRITE('AKAN MEMBUAT LAGI (Y/N) ? ');READLN(LAGI);
    IF (LAGI='Y') OR (LAGI='y') THEN ULANG:=TRUE ELSE ULANG:=FALSE;
    UNTIL ULANG=FALSE;
  END.

```