```
11 [ July ] 24
          Dijkstras algorithm.
 #include < Stolio h >
 # depire INF 9999 8 8
  # dyine MAX 100
Void dijkstra (int c [MAX][MAX],
  of Pintan, interc) L
    int dist EMAXJ, Vis EMAXJ, Count,
    min, u, i,j;
for (1:=1; i <=n; si++) (
       dist [i] = INF;
        Vis CiJ=0;
   digt [src] = 6;
    Vis [SYC] = 0;
     dist [ sre] =0;
      Ving [ Src] =1;
       Count=1;
     While (count 1=n) L
     min = INF;
     for (j=1; j <=n; j++)[
      it (dest Ej] < mingg vis Gj]!=1){
            min = dist [i];
                 U=j 3
            Vis [U] =1.
```

Counttt;

```
for (j=1;j <=n;jtt) (
    it ((min t ccoJCjJ < dist CiJ) 38
 (vi) [] 1121) )PPP
 dist [j] = min + ccostij],
            و الم طوع
  Print ("Shortest dejtonces from source: Id: las
  for (i = 1; 12=n; 1++) L
    17 (dist [i ]== ENI=)
     Print + (11, 1 -) xd: Infinity \n", src, i).
     ene
    Brit (1.d -> 1.d: 1.d ln", sx,i,
           dust CiJ): 3
   int moun () [
    int CEMAXJEMAXJ, n, STC;
Printf ("Entor the number of vertices:");
Print ("Inton the Cost matrix (Entre INF og
             "d): \n", INF);
  for Cint i=1 ;i <= n;i++) (1-0
    for (int J=1; j<=n; j++) L
          Scanf (".1.d", Recidia);
          it (cris GJ == = = = ) 1
              Clit CEIJ LJJ = ENE;
         Prints ("Into the Source vertex: ");
Sont ("Id", Borc);
              votum on
```

output. Entor number of Mentices (max 100). 5 0 3 9999 9999 4 3 0 2 1 9999 9999 2 0 1 9999 4 9999 9999 9999 Grath: Pront (1. d - 1. d: +d/n", sr. i. (O) - - (4) Tran (3) int c [Mxx] max 3, n, src;

Print Mc " Cutor" the mayble of vocices.") Sount (2) - "Total", "Total Trades Edge buight of the 0-12 (+00 (1=2) (1=1/ +n) 1-2 1 (60 (10 = > 1 / - 1 + N) mp 2-4=M3 = = 1 [] (1) Clif cer J 43 - Fare;

```
Lewelcold algorithm
    # include cerdio.n?
    # depire IN= agag
( # dejine 1 [MAX ] 10
   Smuch Edge & wash, wight,
      Struck Subret & CLCI HAI
Distance int Powent; ( ) ( ) = )
             int rance, = 3.2. (1) two
            hught = 0; Culyet (
                (Smut Surget Subgets [], int i).
      void Unionsity (Smuch Subret Subrety EJ, M+x,
         Subject Y to Persont = 1;
         Void keunkols (int C[MAX], EMAX], int n);
   int Find (smut Subset Subsets CJ, int 1) {
  if (Subrety Ci J. Parient !=i)
  Subsety CiJ. Parunt = find Subsety, Subsets CiJ.
        return Subrety (i). Powent;
     void union sety (smeet subrut subrut ( ),
                    int x, inty ) L
              int xroot = find (Subsety, x);
         int ymoot = find (Subrety, y))
    if ( Subuts [xnot] rank < Subrets [ynoot]. rank
Subrets Cxroot J. Parent = Yroot;
    ereif (Subret & [ x root ] Dank > Subrets [most].
       Surets [ x root ]. Parent = x root;
           sue &
```

```
Subjets Cynot J. Parent = x root:
           Subsett [xnob]. rank+++;
    void knuekols (int CEMAXJEMAXJ, int n)
         Street Edge runt [n];
int c=0;
           int isj;
      for (i=0 jizn;itt)
      - aut (i). Src = result [i]. dut = result(i)
           height = 0.
Smit Subert Subert En J.
be Cropien; i++) L
Subuty Cid. Parent =1;
      while (cen = 13 s
 int [min = INF ] U=-1, V=-1;
     for (1:0, 1 cn; 1++) [
  Eliterary Ues = tours ) too nowy brow
  (K. perset) but = 4000x 46
     (CK: Steedys) brit = toony to
      int mem (1 L
int c Emarjemar Jn
Prints ("Suter the number of vortice
      Scont ("-1d", 8n);
```

```
kometals (c, n);
   rown o;
output: Enter the number of vertice; 5
      0 20 60
      2 0 3 8 5
      0 3 0 0 7
      6 8 6 0 9
      0 5 7 9 0
             weight
2
     Edoje
     0-1
     4-2
```