```
# include < stdio. h>
 # include & limits. h>
#define Max-Vertices 100
# define INF INT_MAX
int minky (intn, int dc], int sc]){
     int min = INF, min-kindese.
 for (int v = 0; V < n; V++) {
   if CSCVI == 0 88 d(V) kmin) (
           min = dCVJ;
      min-Indese = v;
   return min index;
```

int Print MST Cint n, int PCJ, int Cost [MAX-VERTICES] (MAX - VERTICES)) L

int total- wst = 0; Printf ["Edge weightin"); tor (int i=1; 1<n; i++) {

Printh (int i=1; icn; i++)(

Print + G.d -1-d "In", Prij, i, cos (i) (Prij) tord-ust += cost [i] [P[i]];

return total-west;

void PrimMST (intn, int cost [MAX_VERTECES] [MAX_VERTELE 5]) {

```
int P [MAX_VERTZCES];
         int & [MAX - VERTICES]
         MIS [ MAX - VERTICES];
       tor Cint i = 0; ien : itt) (
              SCI =0;
   for Cint Count = 0; Count < n-1; Count +t >
         Int U = minkey (n,d,s)
           SCUJ = 1:
    for Cint v= 0; ven; v++) {
 if ( Lost Enjev) 88 SEV) == 088 LOSEV] <
    (Hinling)
     Ders Englisse & DE
         d Cv) = Cost CuJEVJ.
       2147483617 { -1274
    int 605 total-cost = PrimtMST (n, P, cost);
   Brint + (" Total cost of Minimum Stanning Tree(MST)
1. Ed In", total cost );
   int main () ( I to a last of de la Compala) to
       int cost EMAX- VERTICES JEMAX- VERTICES
   Printt l'Enter Dumber of Vertices (max r.d):",
 MAX VERTICES !!
    Scan + ("1.d", 8n).
 Prints Co center the cost adjacency matrix Lucy.d
             bor injinity ): la", INF);
```

for Cint i=0; icn si++) [tor (intj = O'jen , j++) scanf ("t.d", & cost [:][:]); if (cost ci) ci) -- 0 \$8 :1 = 1) [COST CIJ CIJ = INF Printf ("Minimum Spanning Tree Crist) Wing Prim's algorithm: \n"); returno; b, d postan and sail Output: Enter rumber of vertices (max 100): 4 intortu cost adjacency matrix (une 2147483647 borinjinity). 3 4 5 2147483647 90 241183647 2 8 9 0 2147483647 int too total cost = Printmer (n, P) cost): (15 m 10 total ab of Swimmer & worth 1, think carries Tree Crist) living Prim's algorithm Minimum Stanning Eolge Weight. 6-1
2 2147483647 print + latures bromper of Aberto (mars Eq): " Total cost of Minimum Spanning Free (msT): 60 Trans Co with the cost . adjacing Marie Cure V. of

(ans "M: (Graiti)

(0) t -3 (1) -- 1(2) 1 12 - W - W - CI 19 print (" In Drabie: In". Knapsack problem Mamic Pur ramning. # include Estaio, h) int max (inta, intb) 1 return (a; b) a: b word knobsacle (int w, int WESI, int vales, int n)L int j, it; forcizo; ic=n; i++) [tor CUZO, WZ= W wt + DL (E SIL- (C) F FOILW = 20) dpcij [w]=0 Elle 17 CLEI-172=L d ij [i] [w] = max [val[i-1] +dp 10 ments 11-17 [[w-w+ [i-1]], d+ [i-1][w]; of the deli) Costado [1-1][w]; INF ra = dP [n] Lw] annt a Caminimum profit 1-d this, res)] むこりり brint+ ("i'tems included in the knot palle 1 200) for (i=n's i>0 88 res ? 031--) [

```
ij erec = = d Pli-1] b.J.
     Elle L
       Print Cititery 1/d Cweight 1/d Projit xd) 100
         i, 206 [1-1], val [1-1]);
          sy = res-val (1-17.
             w= w- Wet Ci-17.
      printe ("In Drable: In"))
           for Erzo, icansitt)
           for (w=0; W== W; W++)(
           brint ( " dt ; dt ci stw ];
  word knob gade (contract int wells)
      int main (16 200 th)
        int val []= {3,4,5,6};
            int w= 8;
 about [1-17 low ] Kuph gach (ew, wt, val n);
Cally JEI-N return of
: Last ron x am max propit = 10
                 ixeny included in knotsack;
   Itum & lweight: 5 Propt-6)
                  Hund (waght: 3, Projit 4)
by to the municipal in the knowled 1 200).
```

10-1:05 m28 0<1:0=i) rot