WEEK 7

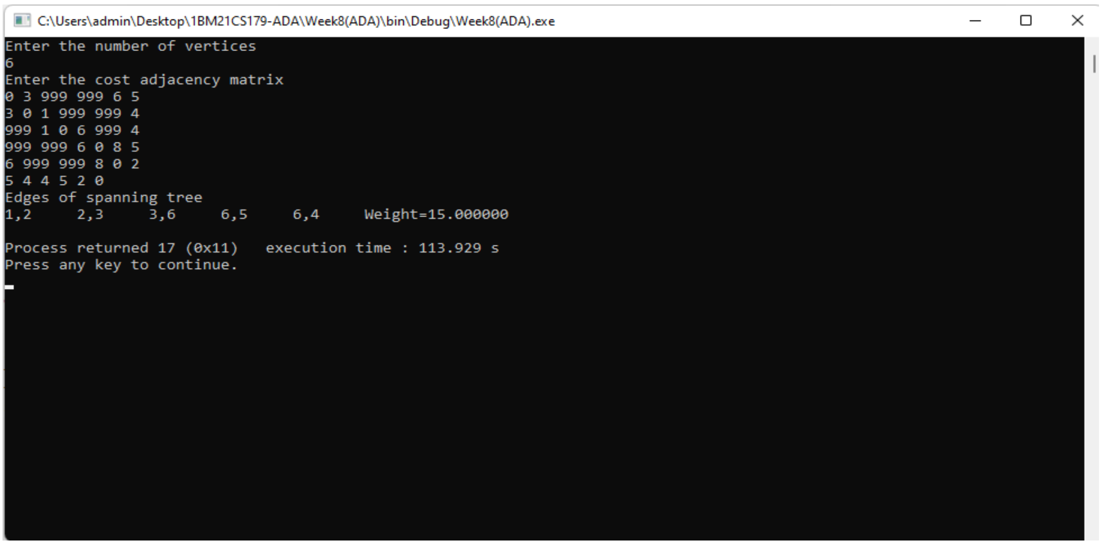
Find Minimum Cost Spanning Tree of a given undirected graph using

Code :

```
#include<stdio.h>
float cost[10][10];
int vt[10],et[10][10],vis[10],j,n;
float sum=0;
int x=1;
int e=0;
void main()
{
   int i;
  printf("Enter the number of vertices\n");
   scanf("%d",&n);
   printf("Enter the cost adjacency matrix\n");
   for(i=1;i<=n;i++)
   {
       for(j=1;j<=n;j++)
       {
             scanf("%f",&cost[i][j]);
       }
       vis[i]=0;
    }
    prims();
    printf("Edges of spanning tree\n");
    for(i=1;i<=e;i++)
    {
         printf("%d,%d\t",et[i][0],et[i][1]);
    }
    printf("Weight=%f\n",sum);

}
void prims()
{
   int s,m,k,u,v;
   float min;
   vt[x]=1;
   vis[x]=1;
   for(s=1;s<n;s++)
   {
       j=x;
       min=999;
       while(j>0)
       {
            k=vt[j];
            for(m=2;m<=n;m++)
            {
              if(vis[m]==0)
              {
                  if(cost[k][m]<min)
                  {
                    min=cost[k][m];
                    u=k;
                    v=m;
                  }
              }
          }
          j--;
    }
    vt[++x]=v;
    et[s][0]=u;
    et[s][1]=v;
    e++;
    vis[v]=1;
    sum=sum+min;
  }
}
```

Output :


C:\Users\admin\Desktop\1BM21CS179-ADA\Week8(ADA)\bin\Debug\Week8(ADA).exe

```
Enter the number of vertices
6
Enter the cost adjacency matrix
0 3 999 999 6 5
3 0 1 999 999 4
999 1 0 6 999 4
999 999 6 0 8 5
6 999 999 8 0 2
5 4 4 5 2 0
Edges of spanning tree
1,2     2,3     3,6     6,5     6,4     Weight=15.000000

Process returned 17 (0x11)   execution time : 113.929 s
Press any key to continue.
```

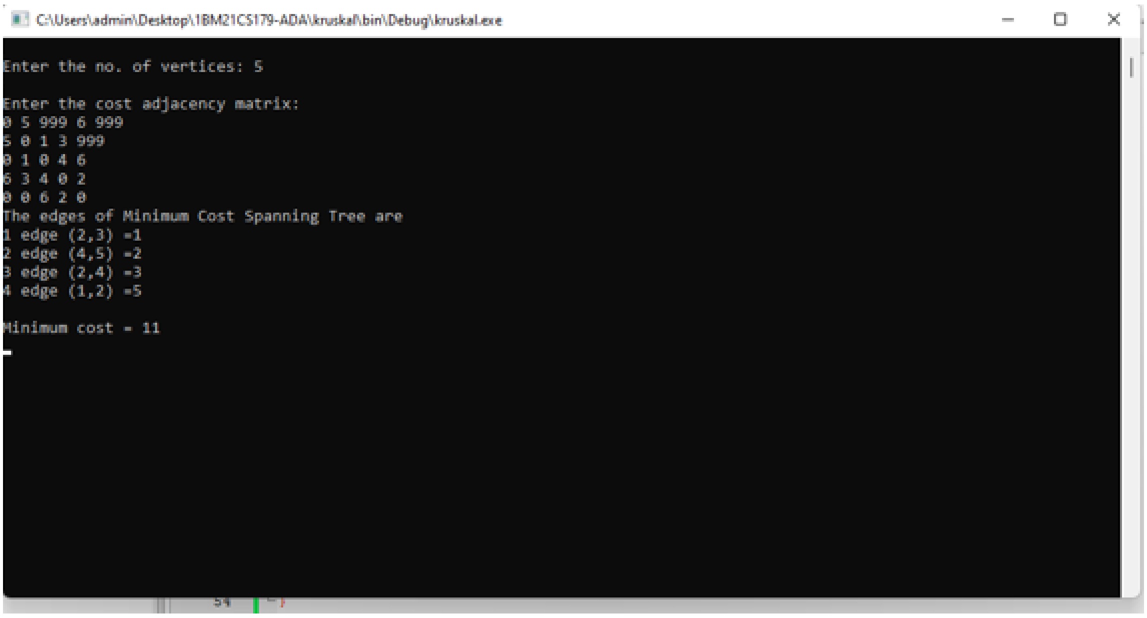
Kruskal’s algorithm

Code :

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
int i,j,k,a,b,u,v,n,ne=1;
int min,mincost=0,cost[9][9],parent[9];
int find(int);
int uni(int,int);
void main()
{
  printf("\nEnter the no. of vertices:");
  scanf("%d",&n);
  printf("\nEnter the cost adjacency matrix:\n");
  for(i=1;i<=n;i++)
  {
    for(j=1;j<=n;j++)
    {
     scanf("%d",&cost[i][j]);
     if(cost[i][j]==0)
     cost[i][j]=999;
    }
  }
  printf("The edges of Minimum Cost Spanning Tree are\n");
  while(ne < n)
  {
    for(i=1,min=999;i<=n;i++)
    {
      for(j=1;j <= n;j++)
      {
        if(cost[i][j] < min)
        {
          min=cost[i][j];
          a=u=i;
          b=v=j;
        }
      }
    }
    u=find(u);
    v=find(v);
    if(uni(u,v))
    {
      printf("%d edge (%d,%d) =%d\n",ne++,a,b,min);
      mincost +=min;
    }
    cost[a][b]=cost[b][a]=999;
  }
  printf("\n\tMinimum cost = %d\n",mincost);
  getch();
}
int find(int i)
```

```
{
   while(parent[i])
   i=parent[i];
   return i;
}
int uni(int i,int j)
{
  if(i!=j)
  {
   parent[j]=i;
   return 1;
  }
  return 0;
}
```

Output :


C:\Users\admin\Desktop\1BM21CS179-ADA\kruskal\bin\Debug\kruskal.exe

```
Enter the no. of vertices: 5

Enter the cost adjacency matrix:
0 5 999 6 999
5 0 1 3 999
0 1 0 4 6
6 3 4 0 2
0 0 6 2 0
The edges of Minimum Cost Spanning Tree are
1 edge (2,3) =1
2 edge (4,5) =2
3 edge (2,4) =3
4 edge (1,2) =5

Minimum cost = 11
```