**Program 15-Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal’s algorithm.**

#include<stdio.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("\n\tImplementation of Kruskal's algorithm\n");

  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);

}

int find(int i)

{

  while(parent[i])

  i=parent[i];

  return i;

}

int uni(int i,int j)

{

  if(i!=j)

  {

    parent[j]=i;

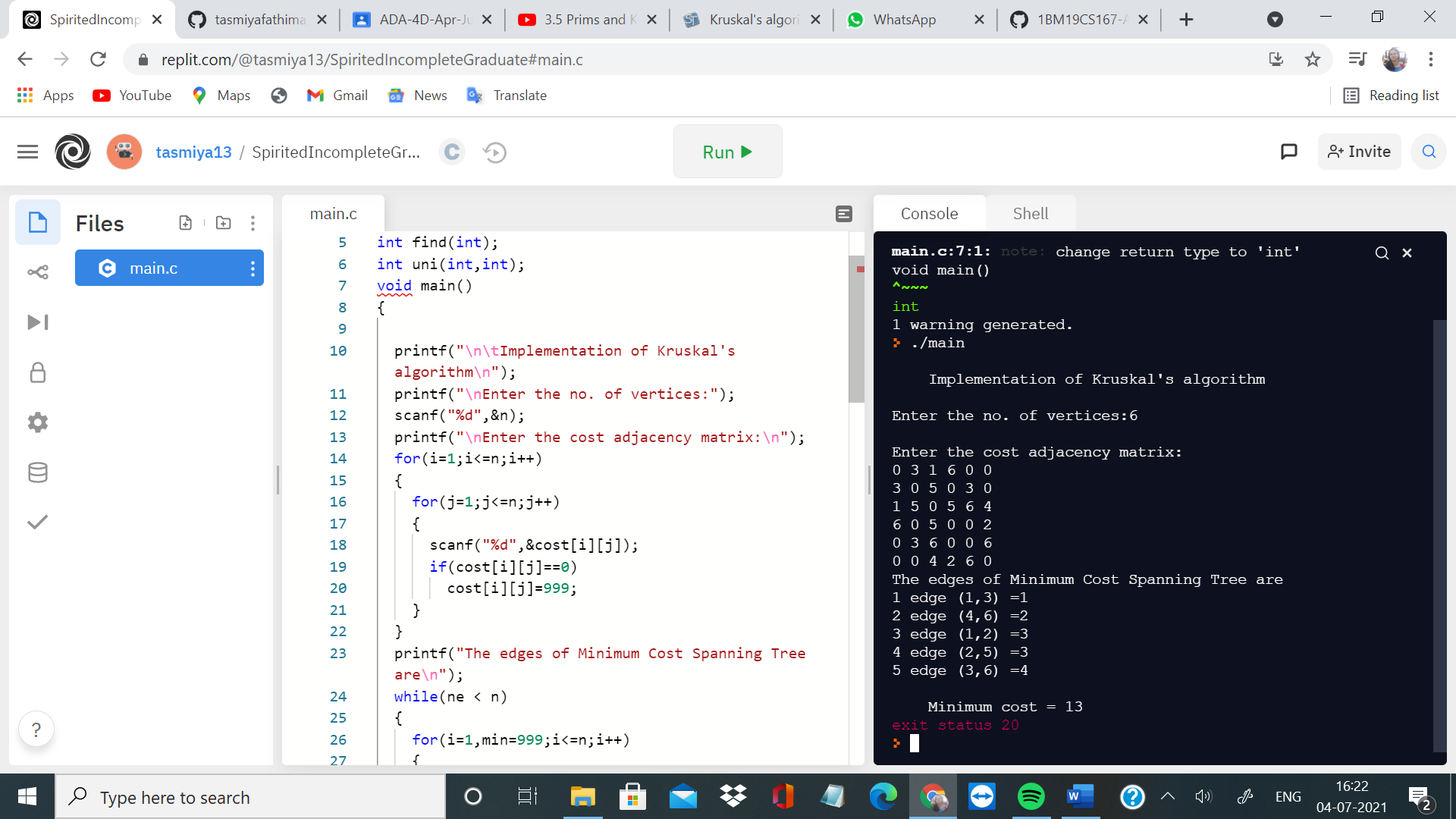
    return 1;

  }

  return 0;

}

**OUTPUT:**

****