**#include <bits/stdc++.h>**

**using namespace std;**

**#define V 5**

**int minKey(int key[], bool mstSet[])**

**{**

**// Initialize min value**

**int min = INT\_MAX, min\_index;**

**for (int v = 0; v < V; v++)**

**if (mstSet[v] == false && key[v] < min)**

**min = key[v], min\_index = v;**

**return min\_index;**

**}**

**void printMST(int parent[], int graph[V][V])**

**{**

**cout << "Edge \tWeight\n";**

**for (int i = 1; i < V; i++)**

**cout << parent[i] << " - " << i << " \t"**

**<< graph[i][parent[i]] << " \n";**

**}**

**void primMST(int graph[V][V])**

**{**

**int parent[V];**

**int key[V];**

**bool mstSet[V];**

**for (int i = 0; i < V; i++)**

**key[i] = INT\_MAX, mstSet[i] = false;**

**key[0] = 0;**

**parent[0] = -1; // First node is always root of MST**

**// The MST will have V vertices**

**for (int count = 0; count < V - 1; count++) {**

**int u = minKey(key, mstSet);**

**mstSet[u] = true;**

**for (int v = 0; v < V; v++)**

**if (graph[u][v] && mstSet[v] == false**

**&& graph[u][v] < key[v])**

**parent[v] = u, key[v] = graph[u][v];**

**}**

**// print the constructed MST**

**printMST(parent, graph);**

**}**

**// Driver's code**

**int main()**

**{**

**int graph[V][V] = { { 0, 2, 0, 6, 0 },**

**{ 2, 0, 3, 8, 5 },**

**{ 0, 3, 0, 0, 7 },**

**{ 6, 8, 0, 0, 9 },**

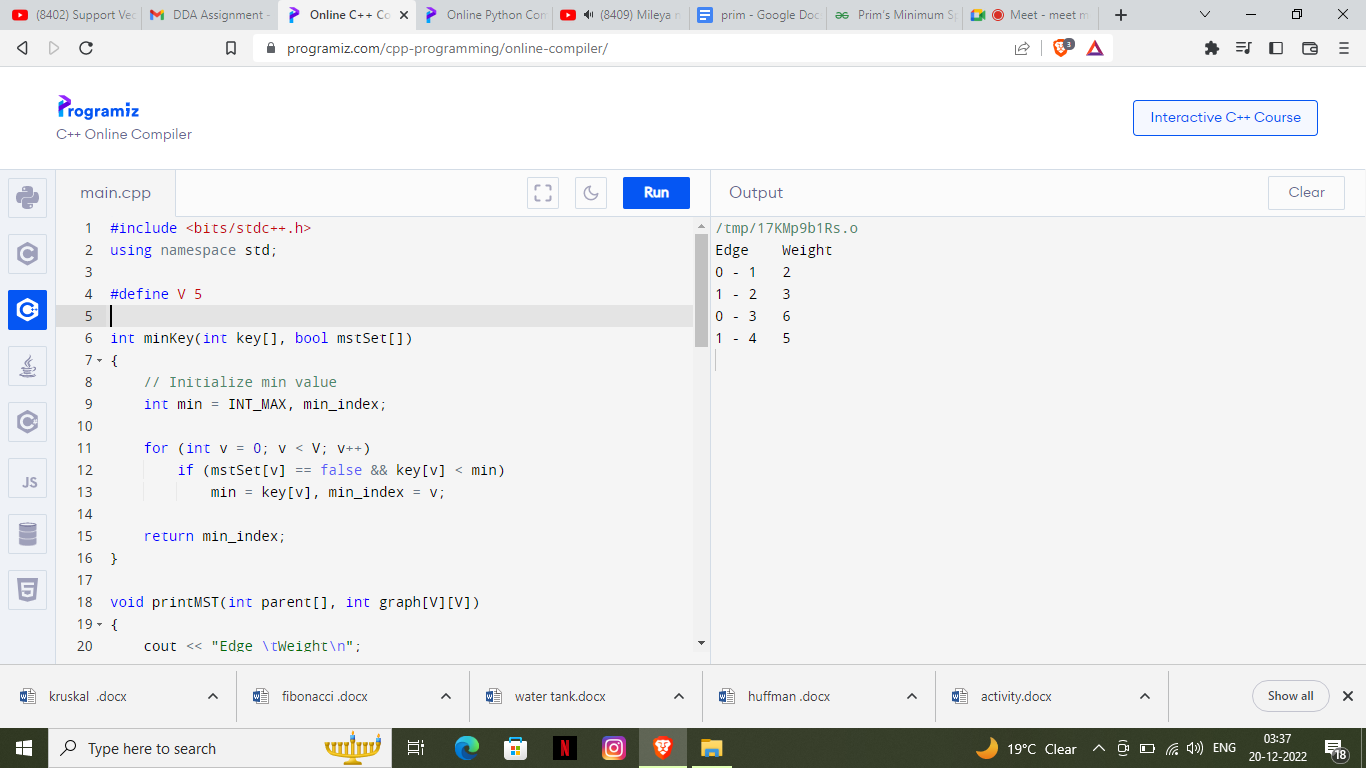
**{ 0, 5, 7, 9, 0 } };**

**// Print the solution**

**primMST(graph);**

**return 0;**

**}**

****