## **DESIGN AND ANALYSIS OF ALGORITHMS – 2CS503**

#### **Practical 5**

Name: Bhanderi Mihir

**Roll No.: 19BCE023** 

Batch No.: A-1

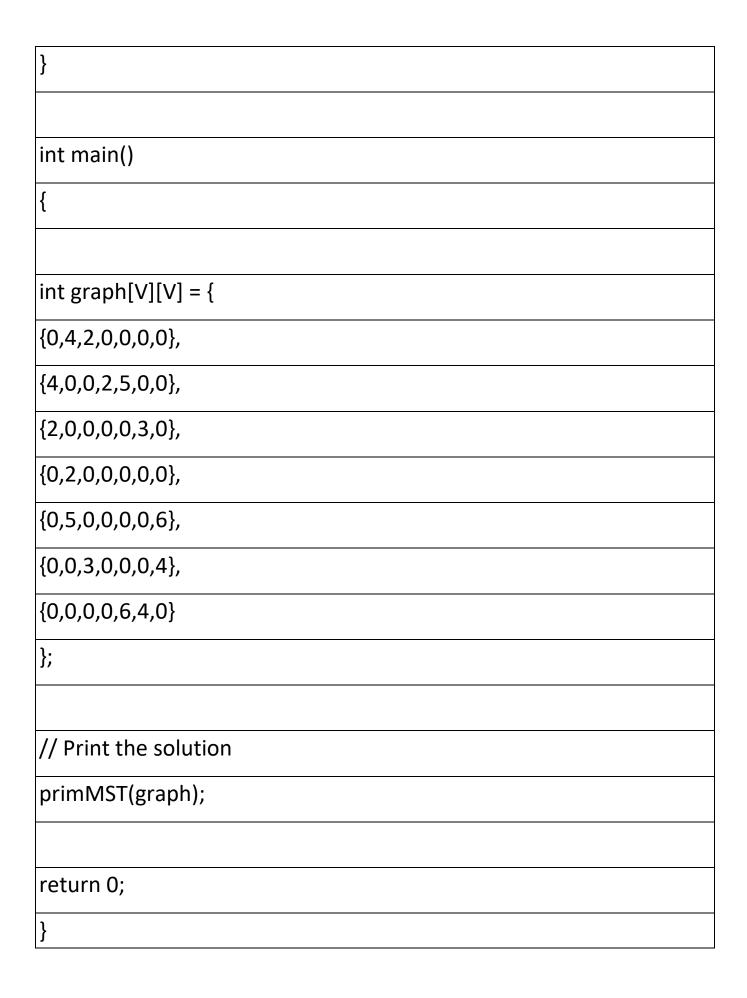
## 1. Prims Algorithm

#### Code:

#include <limits.h></limits.h>				
#include <stdbool.h></stdbool.h>				
#include <stdio.h></stdio.h>				
#define V 7				
int minKey(int key[], bool mstSet[])				
{				
int min = INT_MAX, min_index;				
for (int v = 0; v < V; v++)				

```
if (mstSet[v] == false && key[v] < min)</pre>
min = key[v], min_index = v;
return min_index;
int printMST(int parent[], int graph[V][V])
printf("Edge \tWeight\n");
for (int i = 1; i < V; i++)
printf("%d - %d \t%d \n", parent[i], i, graph[i][parent[i]]);
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;
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];
printMST(parent, graph);
```



# **OUTPUT:**

Edge			Weight		
0	-	1	4		
0	-	2	2		
1	-	3	2		
1	-	4	5		
2	-	5	3		
5	_	6	4		