Lab 7

Prasanna Natarajan
1410110298

Code:

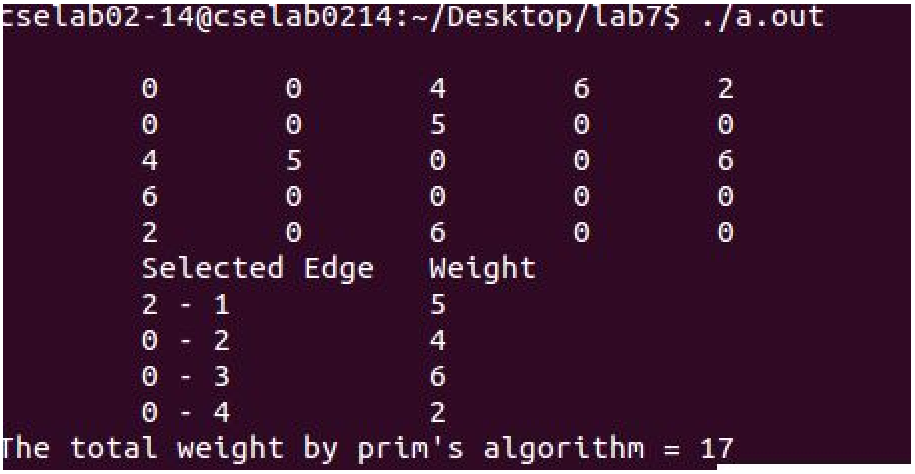
```
/*
Name           :   Prasanna Natarajan
Roll Number    :   1410110298
Inputs         :   A graph stored in adjacency matrix
Outputs        :   A minimum spanning tree with the weight
Method         :   Prim's Algorithm
*/
#include<stdio.h>
#include<stdlib.h>
#include<time.h>

#define N 10 // number of nodes (vertices)
#define M 15 // number of edges
int graph[N][N]; //adjacency matrix
// function declarations
void makeGraph();
void displayGraph();
void prims();
void printMST(int mst[]);
int minKey(int val[], int included[]);
int totalWeight(int mst[]);

int main(){
        time_t t;
        srand((unsigned)time(&t));
        int i,j;
        for(i=0;i<N;i++)
                for(j=0;j<N;j++)
                        graph[i][j] = 0;


        makeGraph();
        displayGraph();
        prims();

        return 0;
}
// function to make a random graph
void makeGraph(){

        int i,r1,r2;

        for(i=0;i<M;i++){
                r1 = rand()%N;
                r2 = rand()%N;
                while(graph[r1][r2] != 0 || r1==r2){
                        r1 = rand()%N;
                        r2 = rand()%N;
                }
                graph[r1][r2] = rand()%10;
                graph[r2][r1] = graph[r1][r2];

        }

}
//function to display the adjacency matrix
void displayGraph(){
        int i,j;
        for(i=0;i<N;i++){
                printf("\n");
                for(j=0;j<N;j++){
                        printf("\t%d ",graph[i][j]);
                }

        }
        printf("\n");
}
//function to print the minimum spanning tree
void printMST(int mst[]){

   printf("\tSelected Edge\tWeight\n");
   int i;
   for (i = 1; i < N; i++)
      printf("\t%d - %d\t\t%d \n", mst[i], i, graph[i][mst[i]]);
}
//function to calculate total weight
int totalWeight(int mst[]){
        int i=0;
        int sum=0;
        for(i=0;i<N;i++)
                sum+=graph[i][mst[i]];

        return sum;

}

// function to choose the minimum index from the not included
int minKey(int val[], int included[])
{
   // Initialize min value
   int min = 100, min_index;
   int j = 0;
   for (j = 0; j < N; j++)
     if (included[j] == 0 && val[j] < min){
        min = val[j];
        min_index = j;
     }

   return min_index;
}


void prims(){

        int mst[N];            // MST
     int val[N];       // values used to pick minimum weight edge in cut
     int included[N];         // To represent set of vertices not yet included in MST

        int i;
        for(i=0;i<N;i++){
                mst[i] = -1;
                val[i] = 100;  // Some relatively large integer
                included[i] = 0;// initialise so that no node in included in the MST
        }

     val[0] = 0;     // picking 0 node as the first node
     mst[0] = -1;    // including 0 node into the MST
     int c = 0;
     for (c = 0; c < N-1; c++){
        int min = minKey(val, included);
        included[min] = 1;
        int j=0;
        for (j = 0; j < N; j++){
          if (graph[min][j] !=0 && included[j] == 0 && graph[min][j] <  val[j]){
             mst[j]  = min;
             val[j] = graph[min][j];
          }
        }
     }

        for(i=1;i<N;i++)
                if(mst[i] ==-1){
                        printf("graph is not connected %d for i = %d\n",mst[i],i);
                        exit(0);
                }

     printMST(mst);
        printf("The total weight by prim's algorithm = %d\n",totalWeight(mst));
}
```

Results:

n	m	Cost for prim's algorithm
5	**5**	17
5	**9**	23
5	**5**	20
10	**15**	31
10	**38**	20
10	**15**	44

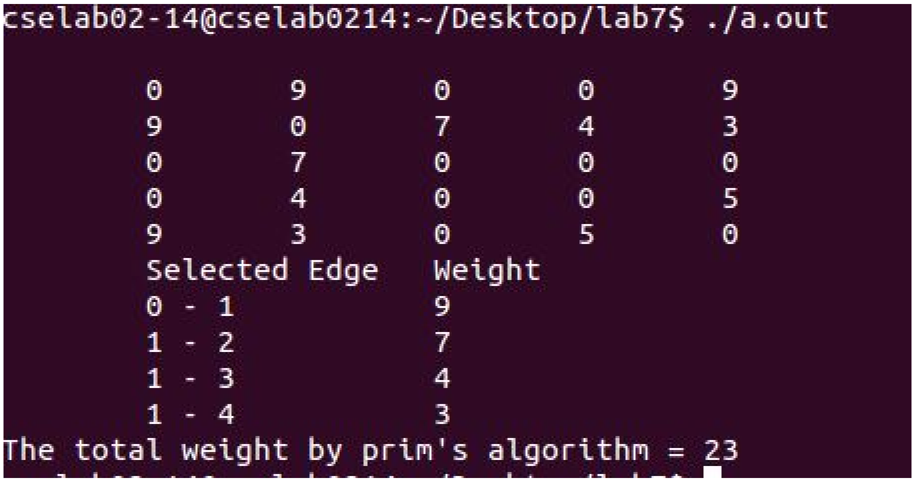
Screenshots:

n=5, m=5 (Case 1)



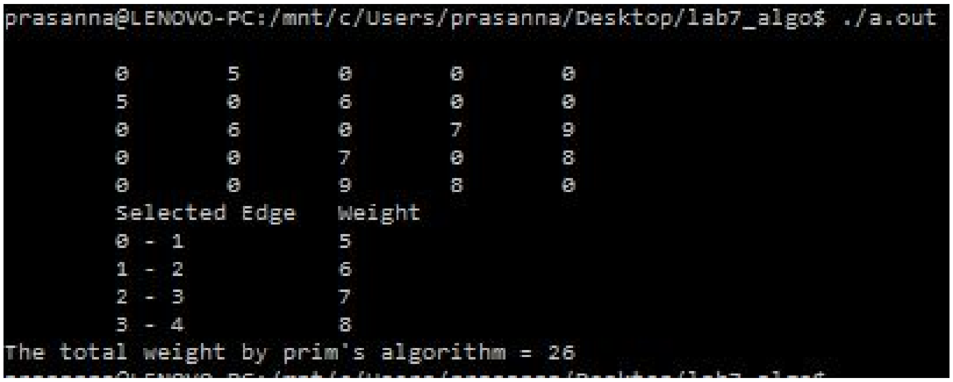
```
cselab02-14@cselab0214:~/Desktop/lab7$ ./a.out

        0       0       4       6       2
        0       0       5       0       0
        4       5       0       0       6
        6       0       0       0       0
        2       0       6       0       0
        Selected Edge   Weight
        2 - 1           5
        0 - 2           4
        0 - 3           6
        0 - 4           2
The total weight by prim's algorithm = 17
```

n=5, m=9 (Case 2)



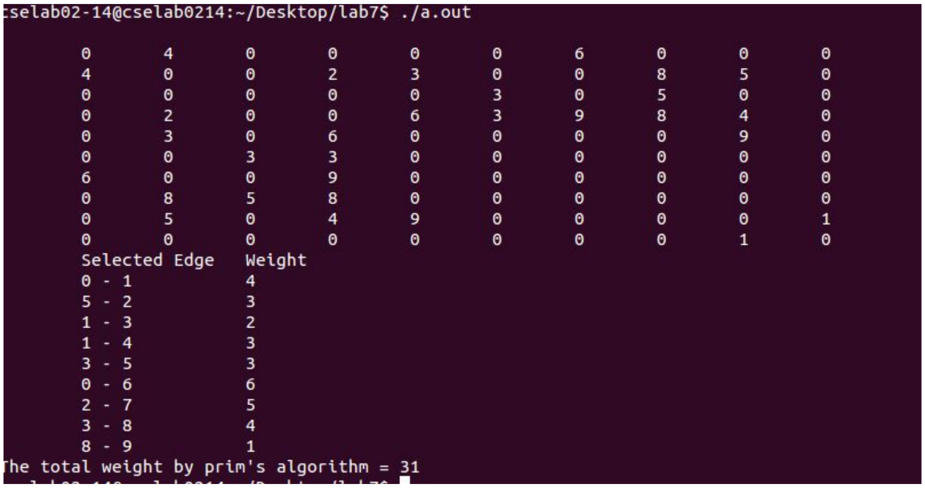
```
cselab02-14@cselab0214:~/Desktop/lab7$ ./a.out

        0       9       0       0       9
        9       0       7       4       3
        0       7       0       0       0
        0       4       0       0       5
        9       3       0       5       0
        Selected Edge   Weight
        0 - 1           9
        1 - 2           7
        1 - 3           4
        1 - 4           3
The total weight by prim's algorithm = 23
```

n = 5, m=5 (Case 3)



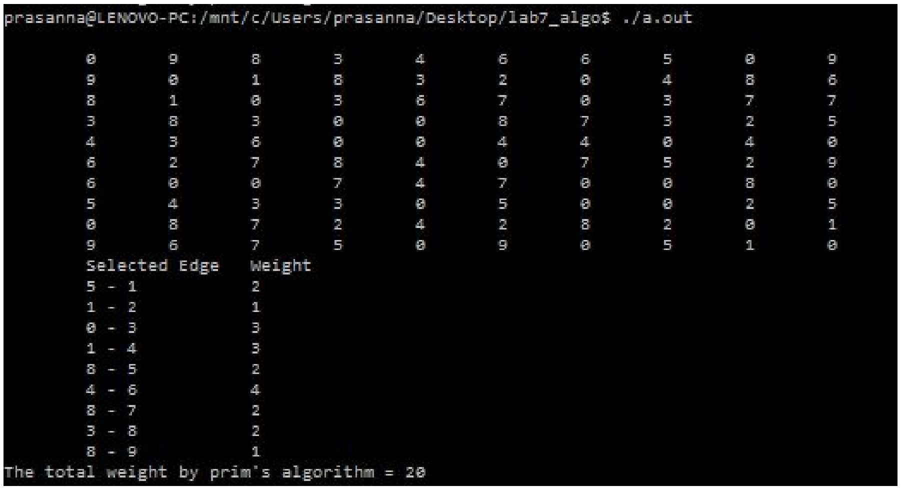
```
prasanna@LENOVO-PC:/mnt/c/Users/prasanna/Desktop/lab7_algo$ ./a.out

        0       5       0       0       0
        5       0       6       0       0
        0       6       0       7       9
        0       0       7       0       8
        0       0       9       8       0
        Selected Edge   Weight
        0 - 1           5
        1 - 2           6
        2 - 3           7
        3 - 4           8
The total weight by prim's algorithm = 26
```

n=10, m=15 (Case 4)



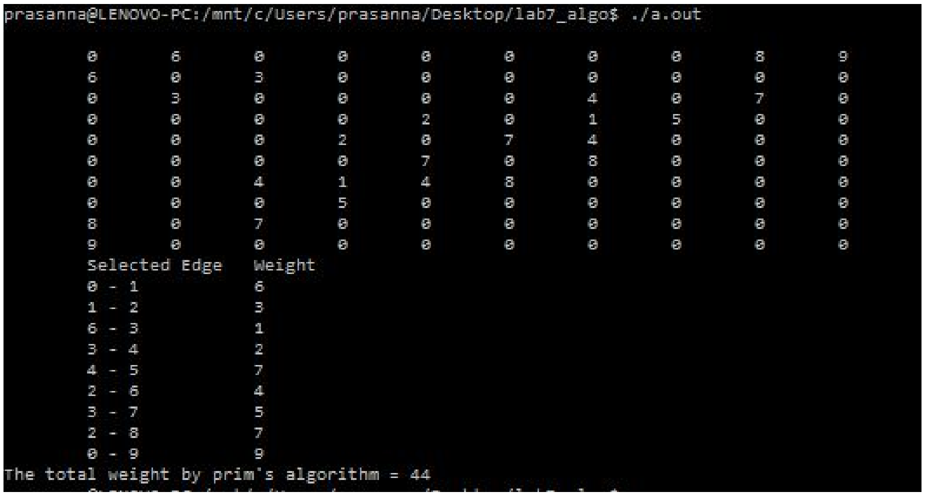
```
:selab02-14@cselab0214:~/Desktop/lab7$ ./a.out

        0       4       0       0       0       0       6       0       0       0
        4       0       0       2       3       0       0       8       5       0
        0       0       0       0       0       3       0       5       0       0
        0       2       0       0       6       3       9       8       4       0
        0       3       0       6       0       0       0       0       9       0
        0       0       3       3       0       0       0       0       0       0
        6       0       0       9       0       0       0       0       0       0
        0       8       5       8       0       0       0       0       0       0
        0       5       0       4       9       0       0       0       0       1
        0       0       0       0       0       0       0       0       1       0
        Selected Edge   Weight
        0 - 1           4
        5 - 2           3
        1 - 3           2
        1 - 4           3
        3 - 5           3
        0 - 6           6
        2 - 7           5
        3 - 8           4
        8 - 9           1
The total weight by prim's algorithm = 31
```

n=10, m=38 (Case 5)



```
prasanna@LENOVO-PC:/mnt/c/Users/prasanna/Desktop/lab7_algo$ ./a.out

        0       9       8       3       4       6       6       5       0       9
        9       0       1       8       3       2       0       4       8       6
        8       1       0       3       6       7       0       3       7       7
        3       8       3       0       0       8       7       3       2       5
        4       3       6       0       0       4       4       0       4       0
        6       2       7       8       4       0       7       5       2       9
        6       0       0       7       4       7       0       0       8       0
        5       4       3       3       0       5       0       0       2       5
        0       8       7       2       4       2       8       2       0       1
        9       6       7       5       0       9       0       5       1       0
        Selected Edge   Weight
        5 - 1           2
        1 - 2           1
        0 - 3           3
        1 - 4           3
        8 - 5           2
        4 - 6           4
        8 - 7           2
        3 - 8           2
        8 - 9           1
The total weight by prim's algorithm = 20
```

n=10, m=15 (Case 6)



```
prasanna@LENOVO-PC:/mnt/c/Users/prasanna/Desktop/lab7_algo$ ./a.out

        0       6       0       0       0       0       0       0       8       9
        6       0       3       0       0       0       0       0       0       0
        0       3       0       0       0       0       4       0       7       0
        0       0       0       0       2       0       1       5       0       0
        0       0       0       2       0       7       4       0       0       0
        0       0       0       0       7       0       8       0       0       0
        0       0       4       1       4       8       0       0       0       0
        0       0       0       5       0       0       0       0       0       0
        8       0       7       0       0       0       0       0       0       0
        9       0       0       0       0       0       0       0       0       0
        Selected Edge   Weight
        0 - 1           6
        1 - 2           3
        6 - 3           1
        3 - 4           2
        4 - 5           7
        2 - 6           4
        3 - 7           5
        2 - 8           7
        0 - 9           9
The total weight by prim's algorithm = 44
```