

Prims

```
def prims(graph, start):  
    visited = set()  
    mst = []  
    edges = []  
  
    visited.add(start)  
  
    for neighbour, weight in graph[start]:  
        edges.append((weight, start, neighbour))  
  
    while edges:  
        edges.sort()  
        weight, u, v = edges.pop(0)  
  
        if v not in visited:  
            visited.add(v)  
            mst.append((u, v, weight))  
  
            for neighbour, w in graph[v]:  
                if neighbour not in visited:  
                    edges.append((w, v, neighbour))  
  
    return mst  
  
graph = {  
    'A': [('B', 1), ('C', 4)],  
    'B': [('A', 1), ('C', 2), ('D', 5)],  
    'C': [('A', 4), ('B', 2), ('D', 1)],  
    'D': [('B', 5), ('C', 1)]  
}
```

```
start = 'A'
tot_cost = 0
span_tree = prims(graph, start)

for source, target, cost in span_tree:
    tot_cost += cost

print("MST using Prim's algorithm is: \n", span_tree)

print("The cost of MST using Prim's algorithm is:", tot_cost)
```