## **Ví dụ**

4 5

1 2 10

2 3 15

1 3 5

4 2 2

4 3 40

⇒ Cây khung nhỏ nhất gồm các cạnh: 1 - 3, 1 - 2, 2 - 4. ⇒ Output: 17

## **Giải thuật:**

B1: Đọc vào danh sách cạnh, chuyển đổi thành danh sách kề.

B2: Dùng thuật toán Prim để tìm cây khung nhỏ nhất.

B3: Xuất kết quả.

## **Mã giả:**

class Node:

index, weight

Prim(start):

dist = [INF] \* N

visited = [False] \* N

pq = PriorityQueue() // dựa trên cái trọng số nhỏ nhất

dist[start] = 0

pq.add(Node(start, 0))

while pq not empty:

u = pq.top().index

visited[u] = True

for (v, w) in graph[u]:

if not visited[v] and dist[v] > w:

dist[v] = w

pq.add(Node(v, w))

res = 0

for i = 0 to N - 1:

if visited[i] == True:

res += dist[i]

return res

main():

read(N, M)

graph = [] // Lưu danh sách đỉnh kề kèm với trọng số của cạnh

for i = 0 to M - 1:

read(u, v, w)

u -= 1

v -= 1

graph[u].append(Node(v, w))

graph[v].append(Node(u, w))

result = Prim(0)

print(result)

Độ phức tạp: Elog(V)