**TASK-1.1 Name: Tanaya Balaji Dahatonde**

**2) Given “n” ropes of different lengths, connect them into a single rope with minimum cost. Assume that the cost to connect two ropes is the same as the sum of their lengths. (Hint: Use Priority Queue)**

#include <iostream>

#include <queue>

#include <vector>

using namespace std;

int findMinCost(vector<int> const &prices)

{

priority\_queue<int, vector<int>, greater<int>> pq(prices.begin(), prices.end());

int cost = 0;

while (pq.size() > 1)

{

int x = pq.top();

pq.pop();

int y = pq.top();

pq.pop();

int sum = x + y;

pq.push(sum);

cost += sum;

}

return cost;

}

int main()

{

vector<int> prices = { 5, 4, 2, 8 };

cout << "The minimum cost is " << findMinCost(prices);

return 0;

}