CSE2218: DATA STRUCTURE AND ALGORITHM II LAB (SUMMER'25)
ASSIGNMENT – 2 TOTAL: 15 MARKS SEC - N

1. **Home Delivery: (5 marks) ->file name: 011xxxxxx_1.cpp**

Mr. Chandler has recently introduced a Home Delivery service at one of his restaurants. Since the service is new, he only employs one delivery person. However, there's a problem—this delivery boy tends to be forgetful!

Every time the delivery boy is sent out for delivery, he forgets to fill up the fuel tank of his scooter beforehand. So, each time Mr. Chandler assigns him a delivery, the boy first stops at the gas station, gets fuel, and then continues to the delivery location. He follows this routine every single time, no matter where the destination is.

To try and deliver the food as quickly as possible, the delivery boy always chooses the shortest path (in terms of time) between the restaurant and the gas station, and then from the gas station to the delivery location.

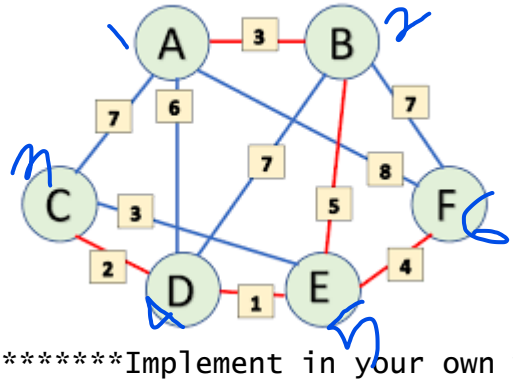
Your task is to help Mr. Chandler figure out how much time the delivery boy could save if he had enough fuel to go directly from the restaurant to the delivery destination without stopping at the gas station. The city has N streets, numbered from 0 to N-1. The restaurant is on street S, the gas station is on street G, and the food has to be delivered to street D. It's possible that S, G, and D might all be the same or different.

Now do the followings:

- Create a class Graph
- Declare an adjacency matrix **adjMat**
- Write a function "**addEdge(u,v,w)**" : which sets adjMat[u][v] = w and adjMat[v][u] = w. This represents the time to travel from street u to street v is w.
- Write a function **findPath(S, D):** which finds the minimum time required to travel from S(restaurant) to D(Delivery location) and also prints the path.
- Write a function **findPath(S, G, D)**: which finds the minimum time required to travel from S(restaurant) to G(Gas station) and then G(Gas station) to D(Delivery location) and also prints the path.

2. **Connect Islands: (5 marks) -> file name: 011xxxxxxx_2.cpp**

You are given a group of islands, and your task is to build bridges between them with the minimum possible cost so that all islands are connected. Each bridge connects two islands, and it has a certain cost to construct. Your objective is to find the minimum total cost required to ensure that every island is connected to at least one other island, either directly or indirectly.
Consider the following graph:



********Implement in your own way. **********
Expected output:
Minimum cost to connect islands: 15
The following bridges should be built:
D->E : 1
C->D : 2
A->B : 3
E->F : 4
B->E : 5

3. **Rod-cutting problem: (5 marks) ->file name: 011xxxxxx_3.cpp**

Given a rod of length n inches and a table of prices pi for i = 1, 2, …, n, determine the maximum revenue obtainable by cutting up the rod and selling the pieces. Note that if the price p_n for a rod of length n is large enough, an optimal solution may require no cutting at all.

Length i		1	3	4	5	7	8
Price pi		2	8	9	10	20	22

Consider the case when n=3, you don't require to cut the rod because selling the entire piece will give the maximum revenue which is 8.
Consider the case when n=5, cutting the rod into three pieces (2 pieces of length 1 and 1 piece of length 3) will give maximum revenue which is p1 + p1 + p3 = 2 + 2 + 8 = 12, which is optimal.
The table may vary. Your task is to solve the rod cutting problem using dynamic programming. Print the maximum revenue and length of each piece.

Submission deadline:
10 Oct, 2025, 11PM
Submit only the 3 cpp files. Don't submit any zip file.
****You will be asked to write pseudocodes as part of the viva. So, write the code on your own and try to understand each line of it. Your assignment will be evaluated based on your written viva****