# Design and Analysis of Algorithms

Fall-2025

Assignment 2

Submission Date: [19-10-2025]

## Instructions:

For each question, Write the Dynamic Approach to solve the problem. Also provide time complexity for each solution. Submit hand written assignments as PDFs. Make sure your solution includes
i) Resulting DP table for given examples
ii) Recurrence relation

### Question 1: Steven's World [10 Marks]

There are N cities and N directed roads in Steven''s world. The cities are numbered from 0 to N - 1. Steven can travel from city i to city (i + 1) % N, ( 0-> 1 -> 2 -> .... -> N - 1 -> 0).

Steven wants to travel around the world by car. The capacity of his car's fuel tank is C gallons. There are a[i] gallons he can use at the beginning of city i and the car takes b[i] gallons to travel from city i to (i + 1) % N.

How many cities can Steven start his car from so that he can travel around the world and reach the same city he started?

**Note**

The fuel tank is initially empty.

**Input Format**

The first line contains two integers (separated by a space): city number N and capacity C.

The second line contains N space-separated integers: a[0], a[1], … , a[N - 1].

The third line contains N space-separated integers: b[0], b[1], … , b[N - 1].

**Output Format**

The number of cities which can be chosen as the start city.

Sample Input

Cities=3 Capacity=3

a[]=[3 1 2]

b[]=[2 2 2]

**Sample Output**

2 (because there exist 2 such cities City0 and City1 which are valid starting positions)

**Explanation**

**First Solution:**

Steven starts from city 0, fills his car with 3 gallons of fuel, and use 2 gallons of fuel to travel to city 1. His fuel tank now has 1 gallon of fuel.

On refueling 1 gallon of fuel at city 1, he then travels to city 2 by using 2 gallons of fuel. His fuel tank is now empty.

On refueling 2 gallon of fuel at city 2, he then travels back to city 0 by using 2 gallons of fuel.

**Second Solution:**

Steven starts from city 2, fill his car with 2 gallons, and travels to city 0.

On refueling 3 gallons of fuel from city 0, he then travels to city 1, and exhausts 2 gallons of fuel. His fuel tank contains 1 gallon of fuel now. He can then refuel 1 gallon of fuel at City 1, and increase his car's fuel to 2 gallons and travel to city 2.

However, Steven cannot start from city 1, because he is given only 1 gallon of fuel, but travelling to city 2 requires 2 gallons.

Hence the answer **2 (because 2 solutions)**.

## Question 2 Gold and Ramesh [10 Marks]

Mr. Ramesh recently stole N grams of gold from ARY Jewellers. He is now on a train back home. To avoid getting caught by the police, he has to convert all the gold he has into paper money. He turns into a salesman and starts selling the gold in the train.

There are passengers who have shown interest in buying the gold. The passenger agrees to buy grams of gold by paying dollars. Ramesh wants to escape from the police and also maximize the profit. Can you help him maximize the profit?

## Note

The passenger would buy exactly grams if the transaction is successful.

## Input Format

The first line contains two space separated integers, and , where P is the number of passengers who agreed to buy and N is the stolen amount of gold (in grams).lines follow. Each line contains two space separated integers - and , where is the the value which the passenger has agreed to pay in exchange for grams of gold.

**Output Format**

If it's possible for Ramesh to escape, print the maximum profit he can enjoy, otherwise print Got caught!.

**Sample Input**

P=4 N=10

460 4

590 6

550 5

590 5

**Sample Output**

1140

Explanation 0

Selling it to passengers buying 4 grams and 6 grams would lead to 1050 dollars whereas selling it to passengers buying 5 grams gold would lead to 1140 dollars. Hence the answer.

**Question 3 [10 Marks]**

You are given a set of coin denominations $\{d_1, d_2, ..., d_n\}$ and a target amount s. Your task is to determine the minimum number of coins required to make the exact sum using the given denominations.

You may assume that:

- There is an infinite supply of each coin denomination.
- It is always possible to make the exact amounts using the given denominations.

**Input:**

Denominations = [1, 3, 4]

Target sum (s) = 6

**Output:**

Minimum coins required = 2

**Question 4 [10 Marks]**

Given N balloons, indexed from 0 to n-1. Each balloon is painted with a number on it represented by an array arr. You are asked to burst all the balloons. If you burst balloon i, you will get arr[left] * arr[i] * arr[right] coins. Here left and right are adjacent indices of i. After the burst, the left and right then become adjacent.

Find the maximum coins you can collect by bursting the balloons wisely

**Question 5 [10 Marks]**
Given two strings **s1** and **s2** and below operations that can be performed on **s1**. The task is to find the **minimum number of edits (operations)** to convert '**s1**' into '**s2**'.

- **Insert**: Insert any character before or after any index of **s1**
- **Remove:** Remove a character of **s1**
- **Replace:** Replace a character at any index of **s1** with some other character.

**Note:** All of the above operations are of equal cost.

---

**Example 1:**

S1: CTATG
S2: TTAAGC.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| C | T | A | T | | | G | |
| | T | | T | A | A | G | C |

**Example 2:**

S1: GACGGATTAG

S2: GATCGGAATAG

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| G | A | | C | G | G | A | T | T | A | G |
| G | A | T | C | G | G | A | A | T | A | G |

**Example 3:**

S1: ATGTG

S2: ACGTA

| | | | | |
|---|---|---|---|---|
| A | T | G | T | G |
| A | C | G | T | A |

**Question 6 [10 Marks]**
A **railway station** has multiple trains arriving and departing at different **times**. A platform is required from the moment a train arrives until it departs. If multiple trains **overlap in timing**, additional platforms are required.
Given two arrays:
**arrivals[]**: Arrival times of N trains.
**departures[]**: Departure times of N trains.
Find the **minimum number of platforms required** so that no train has to wait due to a shortage of platforms.

**Input:**
Arrivals: [18:00, 9:40, 15:00, 11:00, 9:50, 9:00]
Departures: [9:10, 20:00, 19:00, 11:30, 11:20, 12:00]
Time is in 24 hour format
**Output:**
Minimum Platforms Required: 3