National University of Computer & Emerging Sciences, Karachi
Fall-2025 School of Computing (BSCS, BSSE, BSCY, BSAI)
Assignment # 1

Subject: Programming Fundamentals CS-1002	**Post Date: 5/9/2025**
Total Marks: 70	**Due Date:** 19/9/2025
Course Instructors: Javeria Farooq, Sobia Iftikhar, Abeeha Sattar, Saif ur Rehman, Syed Ahmed, Faisal Ali, Fizza Mansoor, Talha Shahid, Abuzar Zafar, Uzma Raza	

Instructions

- There are two parts of assignment. In First part, you will do analysis, designing and implementation of problem solving. In second part you do the dry run and find the output and correct the logic.
- It should be obvious that submitting your work after the due date will result in zero points being awarded.
- Plagiarism (copying/cheating) and late submissions result in a zero mark.

Part 1

Question No. 1

Banks face the challenge of protecting customers from fraudulent activities. Fraud often occurs when someone gains access to another person's account or debit card and begins making unusual purchases. To address this, a bank wants to create a system that can automatically monitor transactions and flag any that appears suspicious.

The system must be able to identify three types of suspicious behavior.

- First, if a customer spends more than a fixed daily limit, the system should consider this unusual. For example, if the daily spending limit is set at 5000, then any transaction that causes the total spending in a day to exceed this amount should be flagged.
- Second, if a transaction occurs in a foreign country that is not commonly used by the customer, the system should also mark it as suspicious. For instance, if the customer usually shops in Pakistan or the UAE, then a transaction in another country such as the USA or UK would be unusual.
- Third, if too many transactions are made within a short period of time, this could also be a sign of fraud. For example, if more than three transactions occur within the same hour, the system should treat the extra transactions as suspicious.

By applying these rules, the bank can provide a first level of protection for its customers. Any suspicious transactions identified by the system can be reviewed by the bank's fraud team for further investigation. This approach allows the bank to reduce risks, protect customer accounts, and build trust by ensuring safer financial operations.

1. You need to do analysis using PAC, IC, and IPO charts.
2. Design an Algorithm, flowchart and pseudocode for the above given case study (You can also use repetition to make the program more effective).
3. Write down the C code to implement your pseudocode (Implement using decisions only, no repetition allowed).

Question No. 2

Assume you are writing a program to help students figure out which quadrant of the graph a point lies in. The quadrants of a graph are the four sections created by the x-axis and y-axis of a Cartesian coordinate plane, each with distinct signs for the x and y coordinates. Starting from the top-right and moving counterclockwise, they are:

- Quadrant I (+x, +y)
- Quadrant II (-x, +y)
- Quadrant III (-x, -y)
- Quadrant IV (+x, -y)

1. You need to do analysis using PAC, IC, and IPO charts.
2. Design an Algorithm, flowchart and pseudocode for the above given problem (You can also use repetition to make the program more effective).
3. Write down the C code to implement your pseudocode (Implement using decisions only, no repetition allowed). The output of your program should be as mentioned below:

> The point (-1, 2) lies in quadrant II
> OR
> The point (2, -3) lies in quadrant IV

Question No. 3

Suppose you are writing a program to help a person figure out if they are eligible for a driving license or not.
An applicant will be asked a series of questions. Based on the answers, it will be decided if the applicant is eligible for a license.

- Prompt the applicant for their age. If the age is 18 or above, ask the next question, otherwise the applicant is ineligible.
- Ask the applicant if they have passed(P) or failed(F) their eyesight exam. If they have passed the test, continue with the next question, otherwise inform them that they might need a prescription for glasses.
- Ask the applicant if they have passed(P) or failed(F) their written test. If they have passed the test, continue with the next question, otherwise inform them that they need to retake the written test.
- Ask the applicant if they have passed (P) or failed(F) their driving test. If they have passed the test, continue with the next question ONLY IF the user is above 60 years old. If they are not above 60, inform them that they are eligible for the license. If they fail the test, then they are not eligible.
- This question is only asked after the applicant has mentioned that they passed their driving test, and if they are over 60 years of age. Ask the applicant if they have a medical fitness certificate or not. If they have the certificate, inform them that they are eligible for the license, otherwise they are not eligible.

1. You need to do analysis using PAC, IC, and IPO charts.
2. Design an Algorithm, flowchart and pseudocode for the above given case study (You can also use repetition to make the program more effective).
3. Write down the C code to implement your pseudocode (Implement using decisions only, no repetition allowed).

Question No. 4
A **poker hand** is represented as a sequence of exactly 5 integers, each in the range **1 to 13**, where:
- **1** represents Ace,
- **2–10** represent cards numbered 2 through 10,
- **11** represents Jack,
- **12** represents Queen,
- **13** represents King.

A hand is said to be a **full house** if it contains:
- **three cards of one rank**, and
- **two cards of a different rank**.

For example, the hand **(1, 2, 1, 2, 1)** is a full house because it contains three cards of rank 1 (Ace) and two cards of rank 2.

1. You need to do analysis using PAC, IC, and IPO charts.
2. Design an Algorithm, flowchart and pseudocode for the above given case study (You can also use repetition to make the program more effective).
3. Write down the C code to implement your pseudocode (Implement using decisions only, no repetition allowed).

Question No. 5
You are tracking how many times the user has entered a number. Assume the user can only enter one-digit numbers (a.k.a 0-9). The user can input numbers as many times as they want, however, if they enter anything that is not a number between 0-9, they will not be allowed to input any more numbers.

After the user can no longer input numbers, your program should display the output in the following format:

Number	Number of Occurrences
0	2
1	10
2	3
3	4
4	8
5	7

1. You need to do analysis using PAC, IC, and IPO charts.
2. Design an Algorithm, flowchart and pseudocode for the above given case study (You can also use repetition to make the program more effective).

Part B

Question No. 5

You are writing a program to keep track of how many times the user has entered a number. Assume the user can only enter one-digit numbers (a.k.a 0-9). The user can input numbers as many times as they want, however, if they enter anything that is not a number between 0-9, they will not be allowed to input any more numbers.

After the user can no longer input numbers, your program should display the output in the following format:

Number	Number of Occurrences
0	2
1	10
2	3
3	4
4	8
5	7

Question No. 6

Consider the following section of C code:

```c
// i, j, and k are ints
if (i < j)
{
        if (j < k)
                i = j;
        else
                j = k;
}
else
{
        if (j > k)
                j = i;
        else
                i = k;
}
printf("%d %d %d\n", i, j, k);
```

What will the code print if the variables i, j, and k have the following values? (Show it with proper dry run).

(a) i is 3, j is 5, and k is 7
(b) i is 3, j is 7, and k is 5
(c) i is 5, j is 3, and k is 7
(d) i is 5, j is 7, and k is 3
(e) i is 7, j is 3, and k is 5
(f) i is 7, j is 5, and k is 3

Question No. 7

The following program attempts to print a message containing an integer representing a day of the week (where 1 = Monday, 2 = Tuesday, …, 7 = Sunday) and then print the corresponding day name. For example: You entered Wednesday. In its current state, the program contains logical errors. Locate the problems and repair them so the program will work as expected. Rewrite the corrected code using switch cases as well.

```c
#include <stdio.h>
int main()
{
        int value;
        printf("Please enter a value in the range 1...5:");
        scanf("%d", &value);
        \\Translate number into its English word
        if (value == 1)
         {
                printf("You entered a one");
         }
else if (value == 2)
{
        printf("You entered a two\n");
}
else if (value == 3)
{
        printf("You entered a three\n");
}
else if (value == 4)
{
        printf("You entered a four\n");
}
else if (value == 5)
 {
        printf("ou entered a five\n");
}
else
{
        printf("You entered a value out of range\n");
}
return 0;
}
```