UNIVERSITI TEKNOLOGI MARA
FINAL TEST

COURSE	:	**FUNDAMENTALS OF ALGORITHMS & COMPUTER PROBLEM SOLVING**
COURSE CODE	:	**CSC126**
EXAMINATION	:	**JANUARY 2025**
TIME	:	**2 HOURS**

INSTRUCTIONS TO CANDIDATES

1. This question paper consists of three (3) parts: PART A (10 Questions)
   PART B (5 Questions)
   PART C (1 Question)

2. Answer ALL questions from all three (3) parts:

   i) Answer PART A in the Objective Answer Sheet.
   ii) Answer PART B and C in the Answer Booklet.

3. Do not bring any material into the examination room unless permission is given by the invigilator.

4. Please check to make sure that this examination pack consists of :

   i) the Question Paper
   ii) an Objective Answer Sheet – provided by the Faculty

5. Answer ALL questions in English.

DO NOT TURN THIS PAGE UNTIL YOU ARE TOLD TO DO SO

This test paper consists of 10 printed pages

PART A (20 MARKS)

1. In an `if...else statement`, the `else` block executes when:

   A. The if condition is true
   B. The if condition is false
   C. Both conditions are true
   D. None of the above

2. Which of the following statements about relational operators is **TRUE**?

   A. The assignment operator = can be used as a relational operator
   B. The expression `6 != 6` evaluates to true
   C. Relational operators are typically used to compare values in selection structures
   D. A condition using the relational operator > can only compare integer values

3. What is the output of the following program segment?

```
#include <iostream>
using namespace std;

int main() {
    int a = 3, b = 7, c = 10, result = 0;

    if (a < b) {
        if (b < c) {
            result = b + c;
            if (result % 2 == 0)
                result /= 2;
            else
                result += a;
        } else {
            result = c - b;
        }
    } else {
        if (a == c) {
            result = c * 2;
        } else {
            result = (a + c) % b;
        }
    }

    cout << "Result: " << result << endl;
    return 0;
}
```


A. Result: 8
B. Result: 18
C. Result: 20
D. Result: 21

4. What is the purpose of curly braces {} in a compound statement?

A. To mark the beginning and end of a program
B. To group multiple statements as a single block
C. To define the scope of variables
D. To make the program run faster

5. What will the following code output if char x = 'G', char y = 'g', char z = 'F'?

```
#include <iostream>
using namespace std;

int main()
{
      char x = 'G', y = 'g', z = 'F';

      if ((x < y) && (z < x))
            cout << "Condition 1";
              else if ((y > z) || (x == 'G'))
                   cout << "Condition 2";
                     else if (z != 'F')
                        cout << "Condition 3";
                          else
                              cout << "Condition 4";
       return 0;
}
```

A. Condition 1
B. Condition 2
C. Condition 3
D. Condition 4

6. Which of the following statements is **FALSE**?

A. The statements in a while loop may never be executed.
B. A loop that executes endlessly is called an infinite loop.
C. It is incorrect to use a for statement within a while loop.
D. A pre-test loop evaluates a condition before executing the loop body.

7. Which of the following statements about sentinel-controlled loops is **TRUE**?

I. The sentinel value is used as the operand in the conditional expression for an indefinite loop.
II. A sentinel-controlled loop is also called a flag-controlled loop.
III. An advantage of sentinel-controlled loops is that you do not need to know how many times the loop will execute.
IV. The while statement is the only type of loop construct that can be used to develop a sentinel-controlled loop.

A. I and II
B. I and III
C. II and IV
D. I and IV

8. What is typically required for a Loop Control Variable (LCV) to function correctly in a loop?

A. It must be declared as a constant
B. It must be initialized after the loop starts
C. It must always increment by 1
D. It must be updated appropriately within the loop body

9. How many times will the following loop be executed?

```
for (int y = 0; y < 8; y = y + 3)
    cout << "Avengers: Endgame";
```

A. 0
B. 2
C. 3
D. 8

10. What will be the output of the following code segment?

```
for (int i = 100; i > 50; i -= 50)
cout << i;
cout << i;
```

A. 10050
B. 1005050
C. 100
D. Error

PART B (35 MARKS)

QUESTION 1

Answer the following questions:

a) Convert the following pseudocode into a C++ program:

```
Start
      If (section == 1) then
            x = x + 5
      Else if (section == 2) then
            x = x + 10
      Else
            Print "Irrational Operation"
      End if
End
```

(2 marks)

```
#include <iostream>

int main() {
      int section;
      int x;
      if (section == 1) {
            x = x + 5
      }
      else if (section == 2) {
            x = x + 10
      }
      else {
            std::cout << "Irrational Operation";
      }
      return 0;
}
```

b) What is the output for the C++program in (a) if section = 5?

(1 mark)

Irrational Operation

c) Convert the C++ program in (a) using the switch statement.

(2 marks)

```
#include <iostream>

int main() {
        int section;
        int x;

        switch (section) {
                case 1: x = x + 5; break;
                case 2: x = x + 10; break;
                default: std::cout << "Irrational Operation";
        }
        return 0;
}
```

QUESTION 2

Yayasan Sarawak offers Kenyalang Scholarship 2025 for Sarawak-born tertiary education students. As a programmer, you are required to write a C++ program that evaluates a student's eligibility for scholarships based on their grade point average (GPA) and annual family income.

The program must take the following inputs:

1. grade point average (GPA)
2. Annual family income

The criteria for the scholarship:

1. Full Scholarship:
    a. GPA is between 3.8 and 4.0.
    b. Family income is less than RM50,000.
2. Partial Scholarship:
    a. GPA is between 3.5 and 3.8.
    b. Family income is less than RM100,000.
3. Need-Based Scholarship:
    a. GPA is between 3.5 and 3.8.
    b. Family income is less than RM50,000.
4. No Scholarship:
    a. Does not meet any of the above criteria.

The program will display the scholarship type for which the student is qualified. It displays one of the following messages based on the input:

- "You got a Full Scholarship." or
- "You got a Partial Scholarship." or
- "You got a Need-Based Scholarship." or
- "No Scholarship."

(10 marks)


```
#include <iostream>
void checkScholarship(float gpa, float familyIncome);

int main() {

        float gpa, familyIncome;

        std::cout << "Yayasan Sarawak Kenyalang Scolarship 2025"
                  << "\nPlease enter your Grade Point Average(GPA): ";
        std::cin >> gpa;


        std::cout << "\nPlease enter your annual family income: RM";
        std::cin >> familyIncome;

        checkScholarship(gpa, familyIncome);
}

void checkScholarship(float gpa, float familyIncome) {
        if ((gpa >= 3.8 && gpa <= 4.0) && familyIncome < 50000) {
                std::cout << "You are eligable for Full Scholarship!";
        }
        else if (gpa >= 3.5 && gpa <= 3.8) {
               if (familyIncome < 100000) {
                        std::cout << "You are eligable for Partial Scholarship!";
               }
               else if (familyIncome < 50000){
                        std::cout << "You are eligable for Need-Based Scholarship!";
               }
               else {
                        std::cout << "You are not eligable for scholarships.";
               }
        }
        else {
                std::cout << "You are not eligable for scholarships.";
        }
}
```

QUESTION 3

Given the following C++ program:

```
#include <iostream>
using namespace std;

int main()
{
      int x = 1;
      int val;
      int min = 21;

      cout << "Enter 5 positive integer numbers from 1 to 20: \n";
      while(x <= 5){
            cout << x << "\t";

            cin >> val;

            if(val % 2 == 0) {
                  if(val < min)
                        min = val;
            }
            x++;
      }
      cout <<"\nThe lowest even number is " << min << endl;

      return 0;
}
```

a) What will be the output when the user inputs the following data into the program above?

```
20    5     6     15    2
```

The lowest even number is 2.

b) Find the values of `x` and `val` after the loop has terminated.
x = 6
val = 2

c) What is the name of the loop control variable in the above program?

x

(5 marks)

QUESTION 4

Consider the following C++ program segment:

```
int num;

cout << "Enter a number: ";
cin >> num;

for(float i = num; i > 0; i++)
{

      if (i / 5 == 0 && i % 2 == 0)
            cout << num << " ";
}
```

The program is supposed to receive a number from the user and display all the odd numbers divisible by 5 in reverse order. Identify the errors and rewrite the program segment to correct them.

(5 marks)

```
#include <iostream>
using namespace std;

int main() {
        int num;

        cout << "Enter a number: ";
        cin >> num;

        for (int i = num; i > 0; i--) {

                if ((num % 5 == 0) && (num % 2 != 0)) {
                        cout << i << " ";
                }
        }
}
```


QUESTION 5

D' Cookies requires a program for their self-order machines to handle customer orders efficiently. The program should allow customers to select their order type: **A** for **Cake** or **B** for **Cookies**. The program should repeatedly ask for the customer's choice until they enter **X** to stop ordering. Once the customer stops, the program displays the total number of cakes and cookies ordered, along with the total price. Each item costs RM10.90, and the total price should be displayed to two decimal places.

The sample input and output of the program are shown below (The shaded texts are the inputs for the program):

```
D'Cookies Self-Order Machine
====================================================
Enter order type (A - Cake, B - Cookies, X - Exit): A
Enter order type (A - Cake, B - Cookies, X - Exit): B
Enter order type (A - Cake, B - Cookies, X - Exit): A
Enter order type (A - Cake, B - Cookies, X - Exit): A
Enter order type (A - Cake, B - Cookies, X - Exit): X

Order Summary:
==============================
Total cakes     : 3
Total cookies   : 1
Total price     : RM43.60
==============================
```

(10 marks)


```
#include <iostream>
#include <cctype>
#include <iomanip>

float calcPrice(int cakes, int cookies);

int main() {
        char orderType = 'A';
        int cakes = 0, cookies = 0;
        float totalPrice = 0;

        std::cout << "D'Cookies Self-Order Machine\n"
                  << "====================================================\n";

        while (orderType != 'X') {
                std::cout << "Enter order type (A - Cake, B - Cookies, X - Exit):
";
                std::cin >> orderType;
                orderType = std::toupper(orderType);

                if (orderType == 'A')
                        cakes++;
                if (orderType == 'B')
                        cookies++;

                totalPrice = calcPrice(cakes, cookies);
        }

        std::cout << std::fixed << std::showpoint << std::setprecision(2);
        std::cout << "\nOrder Summary:\n"
                  << "====================================================\n"
                  << "Total cakes: " << cakes
                  << "\nTotal cookies: " << cookies
                  << "\nTotal price: RM" << totalPrice
                  << "====================================================\n";
}

float calcPrice(int cakes, int cookies) {
        return (cakes + cookies) * 10.90;
}
```

PART C (15 marks)

TypoTees has asked you to write a C++ program to calculate profits from selling **customized water bottles** to multiple customers in bulk. The cost price of each water bottle is fixed at RM15.00. The following table shows the selling price per bottle based on the number of bottles purchased:

Number of Bottles	Price per Bottle (RM)
20 to 29	40.90
30 to 39	35.90
40 and above	30.90

Write a complete C++ program to do the following:

a) Write a function definition for a function named `input()` that allows the users to input the customer's name and the number of water bottles the customer wishes to purchase, and passes the values through its parameters.

(3 marks)

b) Write a function definition for a function named `calcCost()` that receives the number of bottles for each customer and do the following tasks:
   - If the number of bottles is less than 20, display the message "Customer is not allowed to buy less than 20 water bottles!"
   - Otherwise, it calculates and returns the total cost by multiplying the number of bottles purchased by the corresponding selling price.

(8 marks)

c) Write a main() function that will call the functions defined in (a) and (b) to input the customer's name and the number of bottles the customer wishes to purchase, then calculate the total profit obtained. Lastly, the system will display the customer's name, the total cost, and the profit to the customer. Below is sample input and output, where the bold text represents the input.

```
Enter the customer's name: Ahmad Samsudin
Enter the number of water bottles: 27

******PAYMENT DETAILS*****
Customer's Name: Ahmad Samsudin
Total Cost: RM 1104.30
```

(4 marks)


```
#include <iostream>
#include <iomanip>

void input(std::string &customerName, int &nBottles) {
        std::cout << "Enter the customer's name: ";
        std::getline(std::cin, customerName);
        std::cout << "Enter the number of water bottles: ";
        std::cin >> nBottles;
}

float calcCost(int nBottles) {
        if (nBottles < 20) {
                std::cout << "Customer is not allowed to buy less than 20 water
bottles!";
                return 0;
        }
        else {
                if (nBottles >= 20 && nBottles <= 29){
                        return nBottles * 40.90;
                }
                if (nBottles >= 30 && nBottles <= 39){
                        return nBottles * 35.95;
                }
                if (nBottles >= 40){
                        return nBottles * 30.90;
                }
        }
}

int main() {
        std::string customerName;
        int nBottles;
        float totalCost;

        input(customerName, nBottles);

        totalCost = calcCost(nBottles);

        std::cout << "\n******PAYMENT DETAILS*****";
        std::cout << "\nCustomer's Name: " << customerName;
        std::cout << std::fixed << std::setprecision(2);
        std::cout << "\nTotal Cost: RM" << totalCost;

        return 0;
}
```

