Instructions

This assignment is designed to help you practice basic C++ concepts. There are 12 questions in total, including multiple choice questions and coding exercises. The questions aim to improve your understanding of C++ fundamentals like input/output, variables, data types, and selection structures (if, ternary operator, and switch case). Therefore, cheating is forbidden.

I. Multiple choice questions: (Q2, Q4, Q6, Q8, Q10 only will be graded)

1) Which of the following statements is correct about pointers?

a) A pointer can store any type of variable value.
b) A pointer stores the memory address of another variable.
c) Pointers always store integers.
d) Pointers store the size of a variable.

2) What will be the output of the following code?

```
int a = 10, *p;
p = &a;
cout << *p;
```

a) Address of a
b) Value of p
c) 10
d) Garbage value

3) Which of the following statements about recursion is true?

a) Every recursive function must have a loop inside it.
b) Recursive functions can be used to replace any loop.
c) A recursive function must have a base case to prevent infinite recursion.
d) Recursion is generally more efficient than loops for all problems.

4) How do you declare a pointer to an integer in C++?

```
int arr[5] = {1, 2, 3, 4, 5};
int *p = arr;
cout << *(p + 3);
```

a) int *p;
b) int &p;
c) pointer int p;

d) int pointer p;

5) What does the following code do?

a) Outputs the value at index 2 of the array.
b) Outputs the value at index 3 of the array.
c) Outputs the memory address of the array.
d) Outputs the size of the array.

6) What will happen if you dereference a null pointer?

a) The program will execute successfully.
b) The program will crash.
c) It will output 0.
d) It depends on the compiler.

7) Which of the following correctly declares an enum?

a) enum Colors {Red, Green, Blue};
b) enum {Red, Green, Blue};
c) Colors enum {Red, Green, Blue};
d) enum Colors (Red, Green, Blue);

8) Which statement about structures is true?

a) Structures can only store variables of the same type.
b) Structures are automatically private in C++.
c) Structures cannot contain pointers as members.
d) Structures can contain functions and methods in C++.

9) What is the output of the following code?

```
int x = 10, *p = &x;
*p = 20;
cout << x;
```

a) 10
b) 20
c) 0
d) Address of x

10) Which of the following is used to deallocate a pointer in C++?

a) delete
b) free

c) null
d) dealloc

1) What is the size of a pointer on a 64-bit system?

a) 2 bytes
b) 4 bytes
c) 8 bytes
d) Depends on the type of pointer

II. Coding problems: Questions Q1, Q3, Q5, Q7, Q9 only will be graded

1) Create a program to manage an Inventory System:

- Use a structure to represent an item with attributes like itemID, name, quantity, and price.
- Implement functionality to:
    - Add new items to the inventory.
    - Update the quantity of an item using pointers.
    - Display all items in the inventory.

2) Develop a Voting System that uses an enum for candidates (e.g., CandidateA, CandidateB, CandidateC).

- Merhods:
    - Count the votes for each candidate using pointers.
    - Display the results, including the winner.

3) Create a program that manages student records:

- Use a structure to represent a student with attributes like rollNumber, name, and marks.
- Allocate memory dynamically for storing student records using pointers.
- Implement functionality to:
    - Add a new student.
    - Display all students.

4) Write a program that:

- Allocates memory dynamically for an array of integers.
- Allows the user to input values into the array.
- Finds the maximum and minimum values in the array using pointers.

- Deallocates the memory once done.

5) Develop an Address Book program:

   - Use a structure to store contact information (e.g., name, phoneNumber, email).
   - Dynamically allocate memory for multiple contacts using pointers.
   - Provide functionality to:

     - Add a new contact.
     - Search for a contact by name.
     - Display all contacts.

6) Write a program to monitor temperature readings:

   - Use a structure to store city, currentTemp, and maxTemp.
   - Store these readings dynamically in an array of structures using pointers.
   - Allow the user to update the temperature of a specific city.

7) Write a program to dynamically allocate a 2D matrix:

   - Let the user input the size of the matrix.
   - Populate the matrix with values and calculate:

     - The sum of each row.
     - The sum of each column.
     - Deallocate the memory after calculations.

8) Design a Library Book Allocation System:

   - Use a structure for books with attributes like bookID, title, and isAvailable.
   - Dynamically allocate memory for the book list.
   - Implement functionality to:

     - Allocate a book to a user (update isAvailable).
     - Return a book (mark it as available).
     - Display all books.

9) Write a program that:

   - Uses a structure to store patient information (name, age, disease).
   - Dynamically allocates memory for a single patient record using a pointer.
   - Allows the user to:

     - Enter patient details.

- Update the disease of the patient.
- Display the patient's details.

10) Create a program that:

- Uses an enum to represent mathematical operations (Addition, Subtraction, Multiplication, Division).
- Takes two numbers and the desired operation as input.
- Performs the calculation using pointers to pass the numbers to the respective operation and displays the result.

11) Develop a program that:

- Uses pointers to store a temperature value in either Celsius or Fahrenheit.
- Converts the temperature from Celsius to Fahrenheit or vice versa based on user input.
- Displays the converted temperature.

12) Write a program that:

- Uses a structure to represent student data (name, rollNumber, marks).
- Reads student details from the user and writes them to a file using pointers.
- Reads back the stored data from the file and displays it.

13) Create a program that:

- Uses pointers to swap the values of two variables.
- After swapping, displays the updated values of the variables.
- Extend this by asking the user if they want to repeat the operation for new variables.