FINAL TERM

Submitted By:

Hamza Khan Afridi (13071)

BS (SE)

Subject :

Programming Fundamentals

Question1 :

A) What is the purpose of if statement? Discuss its two different forms with examples.

Ans: The IF statement is used to follow a certain set of instructions based on the result of a decision. Consider a situation in real life when you would want to make a decision based on a condition.

```
if (this condition is true )

execute this statement ;
```

IF statement:

The if statement evaluates the CONDITION inside the parentheses().

If the CONDITION evaluates to TRUE the code inside the body of IF is executed.

If the CONDITION evaluates to FALSE the code inside the body of IF is skipped

Example:

```
if (number > 0) {

      cout << "You entered a positive integer: " << number << endl;

  }

  cout << "This statement is always executed.";

  return 0;
```

IF ELSE statement:

if the condition evaluates TRUE, the code inside the body of if is executed,the code inside the body of else is skipped from execution

If the condition evaluates FALSE,the code inside the body of else is executed the code inside the body of if is skipped from execution.

Example:

```
if (number >= 0) {

      cout << "You entered a positive integer: " << number << endl; }

  else {

      cout << "You entered a negative integer: " << number << endl;}

  cout << "This line is always printed.";
```

```
        return 0;}
```

B) Write a C++ program to read two numbers from keyboard and then find the LARGEST number of them.

ANS:

```
#include <iostream>

using namespace std;


int main()

{

   int num1, num2;

   cout<<"Enter first number:";

   cin>>25;

   cout<<"Enter second number:";

   cin>>40;

   if(num1>num2)

   {

        cout<<"First number "<<num1<<" is the largest";

   }

   else

   {

        cout<<"Second number "<<num2<<" is the largest";

   }

   return 0;

}
```

Result the second number 40 is the largest.

Question2 :

A) What are the Logical Operators? Explain them

Ans: Logical Operators: Logical Operators are used to combine two or more conditions/constraints or to complement the evaluation of the original condition in consideration. The result of the operation of a logical operator is a boolean value either true or false.

- && (Logical and) Returns true if both statements are true
- || (Logical or) Returns true if one of the statements is true
- ! (Logical not) Reverse the result, returns false if the result is true

B) Write a C++ program to get Temperature in Fahrenheit ***F*** and then find the Atmosphere according to the below rules:

- If temperature ***F*** is above 40 degree Fahrenheit then display…………………..Very Hot.
- If temperature ***F*** is between 35 & 40 degree Fahrenheit then display………Tolerable.
- If temperature ***F*** is between 30 & 35 degree Fahrenheit then display………Warm.

If temperature ***F*** is less than 30 degree Fahrenheit then display……………..…Cool.

ANS:

```
#include<iostream>

using namespace std;

int main()

{

   float fahr, cel;

   char option;

 cout << "Choose from following option:" << endl;

  cout << "1. Celsius to Fahrenheit." << endl;
   cout << "2. Fahrenheit to Celsius." << endl;
   cin >> option;
 //option for converting celsius into fahernheit
   if (option == '1')
   {
      cout << "Enter the temperature in Celsius: ";
      cin >> cel;
      fahr = (1.8 * cel) + 32.0; //temperature conversion formula
      cout << "\nTemperature in degree Fahrenheit: " << fahr << " F" << endl;
   }
   //option for converting Fahrenheit into Celsius
   else if (option == '2')
   {
      cout << "Enter the temperature in Fahrenheit: ";
      cin >> fahr;
     cel = (fahr - 32) / 1.8; //temperature conversion formula
      cout << "\nTemperature in degree Celsius: " << cel << " C" << endl;
   }
   else
      cout << "Error Wrong Input." << endl;
 return 0;
```

Question3 :

A) What does Looping mean? Explain different loops in C++.

ANS: A loop is used for executing a block of statements repeatedly until a particular condition is satisfied.In C++ we have three types of basic loops:

- **For Loop** : A for loop is similar to a while loop, but streamlines the source code. The for loop statement defines the start and end point as well as the increment for each iteration.
- **While Loop:** A while loop is the simplest form of a programming loop. It states that while a condition is valid, keep looping.the while loop will continue until i is equal to num.
- **Do-while Loop:** It is Like a ‘while’ statement, except that it tests the condition at the end of the loop body.

B) Write a C++ program to read a number from keyboard and then determine whether it is Even or Odd number?

ANS:

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

int main()
{
   int number, remainder;

   cout << "Enter the number : ";
   cin >> number;
   remainder = number % 2;
   if (remainder == 0)
       cout << number << " is an even integer " << endl;
   else
       cout << number << " is an odd integer " << endl;

   return 0;
```

Question4 :

A) What is the purpose of using break and continue statements?

ANS: The purpose of break and continue statement are:

Break statement: There are situations where we want to jump out of a loop instantly, without waiting to get back to the conditional test.The keyword break allows us to do this .When break is encountered inside any loop, control automatically passes to the first statement after the loop. A break is usually associated with an if.The keyword break, breaks the control only from the loop in which it is placed.

Continue statement: The continue statement is used inside loops. When a continue statement is encountered inside a loop, control jumps to the beginning of the loop for next iteration, skipping the execution of statements inside the body of loop for the current iteration.

B) Write a C++ program to find the sum of the following numbers:
1+2+3+………+10

ANS:

```
#include<iostream>

using namespace std;
```

```
int add(int n);


int main()
{
    int n;
     cout << "Enter a positive integer: ";
    cin >> n;
    cout << "Sum =  " << add(n);
     return 0;
}
   int add(int n)
{ if(n != 0)
        return n + add(n - 1);
    return 0;}
```

Question5: What is an array? Explain One-Dimensional and Two-Dimensional Arrays with examples

ANS: An array is a collection of one or more values of the same type. Each value is called an element of the array. The elements of the array share the same variable name but each element has its own unique index number (also known as a subscript). An array can be of any type,

For Example: int, float, char etc. If an array is of type int then it's elements must be of type int only.

One-Dimensional: A one-dimensional array is a structured collection of components (often called array elements) that can be accessed individually by specifying the position of a component with a single index value.

For Example: int number[50]; creates the number array which has 50 components, each capable of holding one int value.

Two-Dimensional Arrays: The elements of a Two-Dimensional array are arranged in rows and columns, and the new operator for Two-Dimensional arrays specifies both the number of rows and the number of columns

For Example: Int [ ] [ ] A; A = new int [3] [4]; This creates a **2D array** of int that has 12 elements arranged in 3 rows and 4 columns.

The end