**Daily Assessment Report**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **22/06/2020** | **Name:** | **Russell D’souza** |
| **Course:** | **C++** | **USN:** | **4al15EC023** |
| **Topic:** | **Module1 : basics concepts** | **Semester & Section:** | **8th &"A" section** |
| **Github Repository:** | **Russell1005** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of session** |
| **Report–Reportcanbetypedorhandwrittenforuptotwopages.**  **1. Module**  Basic concepts of c++  Any entity that has state and behavior is known as an object. For example: chair, pen, table, keyboard, bike etc. It can be physical and logical.  Class  Collection of objects is called class. It is a logical entity.  Inheritance  When one object acquires all the properties and behaviours of parent object i.e. known as inheritance. It provides code reusability. It is used to achieve runtime polymorphism.  Syntax and Structure of C++ program  Here we will discuss one simple and basic C++ program to print "Hello this is C++" and its structure in parts with details and uses.  First C++ program  #include <iostream.h>  using namespace std;  int main()  {  cout <<"Hello this is C++";  }  Header files are included at the beginning just like in C program. Here iostream is a header file which provides us with input & output streams. Header files contained predeclared function libraries, which can be used by users for their ease.  Using namespace std, tells the compiler to use standard namespace. Namespace collects identifiers used for class, object and variables. NameSpace can be used by two ways in a program, either by the use of using statement at the beginning, like we did in above mentioned program or by using name of namespace as prefix before the identifier with scope resolution (::) operator.  Example: std::cout <<"A";  main(), is the function which holds the executing part of program its return type is int.  cout <<, is used to print anything on screen, same as printf in C language. cin and cout are same as scanf and printf, only difference is that you do not need to mention format specifiers like, %d for int etc, in cout & cin.  Comments in C++ Program  For single line comments, use // before mentioning comment, like  cout<<"single line"; // This is single line comment  For multiple line comment, enclose the comment between /\* and \*/  /\*this is  a multiple line  comment \*/  Creating Classes in C++  Classes name must start with capital letter, and they contain data variables and member functions. This is a mere introduction to classes, we will discuss classes in detail throughout the C++ tutorial.  class Abc  {  int i; //data variable  void display() //Member Function  {  cout <<"Inside Member Function";  }  }; // Class ends here  int main()  {  Abc obj; // Creatig Abc class's object  obj.display(); //Calling member function using class object  }  This is how a class is defined, once a class is defined, then its object is created and the member functions are used.  Variables can be declared anywhere in the entire program, but must be declared, before they are used. Hence, we don't need to declare variable at the start of the program.  Don't worry this is just to give you a basic idea about C++ language, we will cover everything in details in next tutorials. |
| Daily Assessment Report   |  |  |  |  | | --- | --- | --- | --- | | Date: | 22/06/20 | Name: | Russell D’souza | | Course: | C++ | USN: | 4AL15EC023 | | Topic: | Module 2 : conditional and loop | Semester & Section: | 8TH & A | | Github Repository: | Russell1005 |  |  |   **AFTERNOON SESSION DETAILS** | |
| **Image of session** | |
| **Report – Report can be typed or hand written for up to two pages.**  1. Module 2  Conditional and loop  In any programming language, loops are used to execute a set of statements repeatedly until a particular condition is satisfied.  How it works    loopflow diagram in C++  A sequence of statement is executed until a specified condition is true. This sequence of statement to be executed is kept inside the curly braces { } known as loop body. After every execution of loop body, condition is checked, and if it is found to be true the loop body is executed again. When condition check comes out to be false, the loop body will not be executed.  There are 3 type of loops in C++ language  1. while loop  2. for loop  3. do-while loop  **while loop**  while loop can be address as an entry control loop. It is completed in 3 steps.  Variable initialization.(e.g int x=0;)  condition(e.g while( x<=10))  Variable increment or decrement (x++ or x-- or x=x+2)  Syntax:  variable initialization;  while (condition)  {  statements;  variable increment or decrement;  }  **for loop**  for loop is used to execute a set of statement repeatedly until a particular condition is satisfied. we can say it an open ended loop. General format is,  for(initialization; condition; increment/decrement)  {  statement-block;  }  In for loop we have exactly two semicolons, one after initialization and second after condition. In this loop we can have more than one initialization or increment/decrement, separated using comma operator. for loop can have only one condition.  **Nested for loop**  We can also have nested for loop, i.e one for loop inside another for loop. Basic syntax is,  for(initialization; condition; increment/decrement)  {  for(initialization; condition; increment/decrement)  {  statement;  }  }  **do...while loop**  In some situations it is necessary to execute body of the loop before testing the condition. Such situations can be handled with the help of do-while loop. do statement evaluates the body of the loop first and at the end, the condition is checked using while statement. General format of do-while loop is,  do  {  // a couple of statements  }  while(condition);  Jumping out of a loop  Sometimes, while executing a loop, it becomes necessary to skip a part of the loop or to leave the loop as soon as certain condition becocmes true, that is jump out of loop. C language allows jumping from one statement to another within a loop as well as jumping out of the loop.  **1) break statement**  When break statement is encountered inside a loop, the loop is immediately exited and the program continues with the statement immediately following the loop.  **2) continue statement**  It causes the control to go directly to the test-condition and then continue the loop process. On encountering continue, cursor leave the current cycle of loop, and starts with the next cycle.  that require a tightly-coupled database where images should be in sync with related data (for example, an employee portal, a stu  you. | |