**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **23/06/2020** | **Name:** | **PRIYA P RAO** |
| **Course:** | **C++ Programming** | **USN:** | **4AL18EC041** |
| **Topic:** | * **Data types, Arrays, Pointers** * **Functions** | **Semester & Section:** | **4th sem ‘A’ section.** |
| **Github Repository:** | **Priya-Rao** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of session**  **C:\Users\Pawan\Desktop\s1.PNG**  **C:\Users\Pawan\Desktop\s2.PNG** |
| **In today’s session I have learnt about:**  **Chapter 1: Data types, Arrays, Pointers**   * + - **Introduction to Data Types :**   **Data types define the type of data a variable can hold, for example an integer variable can hold integer data, a character type variable can hold character data etc. Data types in C++ are categorised in three groups: Built-in, user-defined and Derived.**   * + - **Int, float, double :** * **Int: It short for "integer," is a fundamental variable type built into the compiler and used to define numeric variables holding whole numbers.** * **Float: It is a shortened term for "floating point." By definition, it's a fundamental data type built into the compiler that's used to define numeric values with floating decimal points.** * **Double: The double is a fundamental data type built into the compiler and used to define numeric variables holding numbers with decimal points.**   + - **String, char, bool :** * **String: The term string generally means an ordered sequence of characters, with a first character, a second character, and so on, and in most programming languages such strings are enclosed in either single or double quotes. In C++ the enclosing delimiters are double quotes.** * **Char: It is defined by C++ to always be 1 byte in size. By default, a char may be signed or unsigned. If you're using chars to hold ASCII characters, you don't need to specify a sign.** * **Bool: In C++, the data type bool has been introduced to hold a boolean value, true or false. The values true or false have been added as keywords in the C++ language. Important Points: The default numeric value of true is 1 and false is 0.**   + - **Variable naming rules :**   **Variable names in C++ can range from 1 to 255 characters. All variable names must begin with a letter of the alphabet or an underscore(\_). After the first initial letter, variable names can also contain letters and numbers. Variable names are case sensitive.**   * + - **Arrays :**   **An array is a collection of elements of the same type placed in contiguous memory locations that can be individually referenced by using an index to a unique identifier. Five values of type int can be declared as an array without having to declare five different variables.**   * + - **Using arrays in loops:**   **We can loop through the array elements with the for loop. The following example outputs all elements in the cars array.**  **Example: string cars[4]={“Volvo”, “BMW”, “Ford”, “Mazda”};**  **for(int i=0; i<4; i++)**  **{**  **cout << cars [i] << “\n”;**  **}**   * **Arrays in Calculations :**   **In C++, an array is a variable that can store multiple values of the same type.  C++ Array Declaration**   * **int - type of element to be stored.** * **x - name of the array.** * **6 - size of the array.**   + - **Multi-Dimensional Arrays :**   **In C++, we can define multidimensional arrays in simple words as array of arrays. Data in multidimensional arrays are stored in tabular form.**   * + - **Introduction to Pointers :**   **C++ allows you to have pointer on a pointer and so on. Passing an argument by reference or by address both enable the passed argument to be changed in the calling function by the called function. C++ allows a function to return a pointer to local variable, static variable and dynamically allocated memory as well.**   * + - **More on Pointers :**   **Pointers are said to "point to" the variable whose address they store. An interesting property of pointers is that they can be used to access the variable they point to directly. This is done by preceding the pointer name with the dereference operator ( \* ). The operator itself can be read as "value pointed to by".**   * + - **Dynamic memory :**   **Dynamic memory allocation in C++ refers to performing memory allocation manually by programmer. Dynamically allocated memory is allocated on Heap and non-static and local variables get memory allocated on Stack**   * + - **The sizeof() Operator :**   **sizeof is a unitary operator in the programming languages C and C++. It generates the storage size of an expression or a data type, measured in the number of char-sized units. Consequently, the construct sizeof (char) is guaranteed to be 1.**  **Chapter 2: Functions**   * **Introduction to Functions :**   **A function is a block of code that performs a specific task. It has a name and it is reusable i.e. it can be executed from as many different parts as required. It also optionally returns a value to the calling function.**   * **Function Parameters :**   **A function parameter (sometimes called a formal parameter) is a variable declared in the function declaration.**   * **Functions with Multiple Parameters :**   **Inside the function, you can add as many parameters as you want**  **Note that when we are working with multiple parameters, the function call must have the same number of arguments as there are parameters, and the arguments must be passed in the same order.**   * **The rand() Functions :**   **With the C++ rand() method, you can return a positive number within the range from 0.0 to RAND\_MAX . If you want this function to work, you need to add the <cstdlib> header.**   * **Default Argument :**   **A default argument is a value provided in a function declaration that is automatically assigned by the compiler if the caller of the function doesn't provide a value for the argument with a default value. Following is a simple C++ example to demonstrate the use of default arguments.**   * **Function Overloading :**   **Function overloading is a C++ programming feature that allows us to have more than one function having same name but different parameter list, when I say parameter list, it means the data type and sequence of the parameters**   * **Recursion :**   **The process in which a function calls itself is known as recursion and the corresponding function is called the recursive function. The popular example to understand the recursion is factorial function. Factorial function: f(n) = n\*f(n-1), base condition: if n<=1 then f(n) = 1.**   * **Passing Arrays to Functions :**   [**Arrays**](https://www.programiz.com/cpp-programming/arrays)**can be passed to a**[**function**](https://www.programiz.com/cpp-programming/function)**as an argument. Consider this example to pass one-dimensional array to a function**   * **Pass by Reference with Pointers :**  1. **In pass-by-value, a clone is made and passed into the function.** 2. **In pass-by-reference, a pointer is passed into the function.** 3. **In pass-by-reference with reference arguments, you use the variable name as the argument.** |