**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **18-06-2020** | **Name:** | **Kavya M M** |
| **Course:** | **C programming** | **USN:** | **4AL17EC040** |
| **Topic:** | 1. **Basic concept** 2. **Conditionals and loops** 3. **Function, array and pointers** 4. **Strings and function pointers** | **Semester & Section:** | **6th A** |
| **Github Repository:** | **Kavya\_ECE040** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
|  |
| Basic concepts:   * C programming basic commands to write a C program * A simple C program with output and explanation * Steps to write C programs and get output * Creation, compilation and execution of C program * How to install C compiler and IDE tool to run C programming codes * Basic structure of C program * Example C program to compare all the section * Description for each section of the C program * C programming with definition and output- C programming for prime number, factorial, Fibonacci series, palindrome, swapping 2 numbers with and without temp variable, sample calculator program and sample bank application program etc * If you want to create, compile and execute C programs by your own, we have to install C compiler in our machine. Then , we can start to execute our own C programs in our machine   Conditionals and loops:   1. **If statements:**  * This is the simplest form of the branching statements * It takes an expression in parenthesis and an statement or block of statements, if the condition is true the statements or block of statements get executed otherwise statements are skipped  1. **Switch statement:**  * The switch statements is much like a nested if…else statement. Its mostly a matter of preference which you use, switch statements can be slightly more efficient and easier to read  1. **Using break keyword:**  * If a condition is met in switch case then execution continues on into the next case clause also if it not explicitly specified that the execution should exit the switch statement, this is achieved by using break keyword  1. **While loop:**  * The most basic loop in C is the while loop. A while statement is like a repeating if statement. Like an if statement, if the test condition is true: the statements get executed. The difference is that statement have been executed, the test condition is checked again. If it is still true the statement gets executed again. This cycle repeat until the test condition evaluates to false  1. **For loop:**  * For loop is similar to a while, it’s just written in different way. For statement are often used to process lists such as a range of numbers  1. **Do…while loop:**  * It is just a while loop except that the test condition is checked at the end of the loop rather than at the start. This has the effect that the content of the loops are always executed at least once  1. **Break and continue statement:**  * C provides 2 commands to control how we loop * Break-exit from loop or switch * Continue- skip 1 iteration of loop   Functions, arrays and pointers:  **Functions:**   * A function is a group of statements that together perform a task. Every C program has at least 1 function, which is main (), and all the most trivial programs can define additional functions * We can divide up code into separate functions. How we divide up code among different functions is up to us, but logically the division is such that each function performs specific task * The C standard library provides numerous built-in functions that the program can call. * Example: strcat(),-to concatenate 2 string   memcpy()- to copy 1 memory location to another location and etc  **Array:**   * Array a kind of data structure that can be store a fixed size sequential collection of elements of the same type. An array is used to store a collection of data, but it often more useful to think of an array as a collection of variables of the same type * All array consists of continuous memory locations. The lowest address corresponds to the 1st elements and the highest address for last element   **Pointer:**   * A pointer is a variable whose value is the address of another variable, i.e., direct address of the memory location.. like any variable or constant we must declare a pointer before using it to store any variable address   String and function pointer  **String:**  String is actually 1 dimensional array of character terminated by a null character ‘\0’. Thus, a null-terminated string contains the characters that comrise the string followed by a null |