DAILY ASSESSMENT FORMAT

Date:	18 June 2020	Name:	Karegowda kn
Course:	C programming	USN:	4al16ec029
Topic:	Basic concepts, Conditionals & Loops	Semester & Section:	6 sem B sec
Github Repository:	Karegowda-courses		

FORENOON SESSION DETAILS

Report--

Basic concept of c

C is a procedural programming language. It was initially developed by Dennis Ritchie as a system programming language to write operating system. The main features of C language include low-level access to memory, simple set of keywords, and clean style, these features make C language suitable for system programming like operating system or compiler development.

C For loop

This is one of the most frequently used loop in C programming.

Syntax of for loop:

```
for (initialization; condition test; increment or decrement)
 {       //Statements to be executed repeatedly
 }
```

Example of For loop

```
#include <stdio.h>
int main()
    {
         int i;
        for (i=1; i<=3; i++)
          {
               printf("%d\n", i);
          }
         return 0;
   }
```

Output:

```
1
2
3
```

Nested For Loop in C :

Nesting of loop is also possible. Lets take an example to understand this

```
#include <stdio.h>
 int main()
       {
             for (int i=0; i<2; i++)
             {
                   for (int j=0; j<4; j++)
                     {
                          printf("%d, %d\n",i ,j);
                     }
              }
             return 0;
        }
```

Output:

```
0, 0
0, 1
0, 2
0, 3
1, 0
1, 1
1, 2
 1, 3
```

Multiple initialization inside for Loop in C

We can have multiple initialization in the for loop as shown below.

for (i=1,j=1;i<10 && j<10; i++, j++)

What’s the difference between above for loop and a simple for loop?

1. It is initializing two variables. Note: both are separated by comma (,).

2. It has two test conditions joined together using AND (&&) logical operator. Note: You cannot use multiple test conditions separated by comma, you must use logical operator such as && or || to join conditions.

3. It has two variables in increment part. Note: Should be separated by comma.

Example of for loop with multiple test conditions

```
#include <stdio.h>
 int main()
{
      int i,j;
     for (i=1,j=1 ; i<3 || j<5; i++,j++)
        {
             printf("%d, %d\n",i ,j);
        }
        return 0;
 }
```

DAILY ASSESSMENT FORMAT

Date:	18 June 2020	Name:	Karegowda kn
Course:	C programming	USN:	4al16ec029
Topic:	Functions, Array & Pointers, Strings & Function Pointers	Semester & Section:	6 sem B sec
Github Repository:	Karegowda-courses		

AFTERNOON SESSION DETAILS

Report--

There are two types of functions in C programming:

Library Functions: are the functions which are declared in the C header files such as scanf(), printf(), gets(), puts(), ceil(), floor() etc.

User-defined functions: are the functions which are created by the C programmer, so that he/she can use it many times.

Array

An array is a collection of data items, all of the same type, accessed using a common name. A onedimensional array is like a list; A two dimensional array is like a table; The C language places no limits on the number of dimensions in an array, though specific implementations may.

C – Pointer

Pointers in C language is a variable that stores/points the address of another variable. A Pointer in C is used to allocate memory dynamically i.e. at run time. The pointer variable might be belonging to any of the data type such as int, float, char, double, short etc.

Function Pointer in C

In C, like normal data pointers (int *, char *, etc), we can have pointers to functions. Following is a simple example that shows declaration and function call using function pointer.

filter_none

edit

play_arrow

brightness_4

```
 #include <stdio.h>
// A normal function with an int parameter
// and void return type
void fun(int a)
```

```
{
    printf("Value of a is %d\n", a);
}

int main()
{
    // fun_ptr is a pointer to function fun()
    void (*fun_ptr)(int) = &fun;

    /* The above line is equivalent of following two
       void (*fun_ptr)(int);
       fun_ptr = &fun;
    */

    // Invoking fun() using fun_ptr
    (*fun_ptr)(10);

    return 0;
}
```

Output: Value of a is 10

String

In C programming, a string is a sequence of characters terminated with a null character \0 . For example: char c[] = "c string"; When the compiler encounters a sequence of characters enclosed in the double quotation marks, it appends a null character \0 at the end by default.