

# Introduction to Programming

## Assignment 02

*Due Date*

**Softcopy submission:- 17-Nov-2022; 23:59 hrs**

**Hardcopy submission:- 18-Nov-2022; 13:00 hrs**

### ***Instructions:***

1. Write the assignment on A4 sheets.
2. Write both questions and answers in the assignment. It should be handwritten.
3. Clearly mention your Name, Registration Number, Semester, School and email ID at the beginning of the assignment sheet.
4. After you have completed the assignment, scan the assignment sheets as a single PDF, preferably using Adobe Scanner in mobile, and submit the softcopy of the assignment as PDF in the LMS within the Softcopy Submission deadline.
5. Submit the physical paper copy of the assignment on Hardcopy Submission deadline with multiple papers properly stapled.
6. The assignments will be graded only for those who have submitted both softcopy and hardcopy within the due date.
7. **There will not be any extension to the submission deadline.**

### ***Note:***

- For Part-B, write a detailed explanation behind how the code works.
  - For Part-C, write the logic or the steps you will use to solve the problem in the assignment sheet.
  - You have to submit a working source code for Part-C.
    - No need to write the source code in the assignment sheet.
  - So, you will have to make two submissions in LMS
    - The PDF softcopy of the written part
    - C Source code for Part-C
-

Part A  
(10 x 1 = 10)

1. Out of the following, which one is not a valid keyword to define a data type for a variable?

- a) int
- b) float
- c) real
- d) double

2. In terms of memory usage by various C variable types, find the correct order.

- a) char > int > float
- b) char < int < double
- c) double > char > int
- d) int > char > float

3. Which of the following is not an operator in C?

- a) ++
- b) !
- c) sizeof()
- d) None of the above

4. Which one of the following is not a valid identifier?

- a) cat1
- b) c\_a\_t\_1
- c) 1cat
- d) \_cAt1

5. What is the equivalent C statement for the following?  $x -= y + 1$ ;

- a)  $x = x - y - 1$ ;
- b)  $x = x - y + 1$ ;
- c)  $x = -x - y - 1$ ;
- d)  $x = x + y - 1$ ;

6. What is the correct order of evaluation for the expression

$$z = x + y * z / 4 \% 2 - 1;$$

- a)  $* / \% = + -$
- b)  $* / \% + - =$
- c)  $/ * \% - + =$
- d)  $- + = * \% /$

7. If addition had higher precedence than multiplication, what will be the value of the following expression:

$$x = 1 + 2 * 3 + 4 * 5;$$

- a) 27
- b) 47
- c) 69
- d) 105

8. How many times will the following loop execute?

```
for (j = 1; j <= 10; j = j-1)
```

- a) Forever
- b) Never
- c) 0
- d) 1

9. If no return type is mentioned in the function declaration, what is the default data type assumed?

- a) void
- b) int
- c) float
- d) char

10. Missing elements of partially initialized integer arrays are filled with?

- a) NULL
- b) 0
- c) Garbage values
- d) 1

---

### Part B

( 6 x 4 = 24)

*For all questions in Part B, apart from writing the output, you should also write a detailed explanation of the logic behind why that particular output is obtained.*

11. Explain with output

```
int main(void)
{
    int value = 3;
    value = value + 2 * value++;
    printf("%d \n", value);
    return 0;
}
```

**12. Explain with output**

```
int main(void)
{
    int value;
    value = 5;
    while(value <= 10)
    {
        printf("%d ",value);
        if(value > 7)
            break;
        value++;
    }
    printf("%d \n",value+10);
    return 0;
}
```

**13. Explain with output**

```
int main(void)
{
    int value = 1;
    while(value++ <= 1)
    {
        while(value++ <= 2)
            printf("%d ", value);
    }
    printf("%d ",value);
    return 0;
}
```

**14.** Explain with output

```
int main(void)
{
    int value;
    value = 10;
    do
    {
        ;
    } while(value++ < 10);
    while(value++ <= 11);
    printf("%d",value);
    return 0;
}
```

**15.** For the following code, write an equivalent code using while loop instead of for loop. Explain how this equivalence is established.

```
int factorial (int n)
{
    int i , ret = 1 ;
    for (i = 2 ; i <= n ; i ++ )
        ret *= i ;
    return ret;
}
```

**16.** Explain the following code with correct output:

```
int main(void)
{
    float a[] = {4.3, 3.2, 2.6, 6.4, 1.9};
    printf("%ld bytes\n", sizeof(a));

    return 0;
}
```

---

### Part C

( 1 x 20 = 20)

**17.** Bus fare calculator:

You are provided with an array of characters

```
stops = ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H']
```

Assume the distance between two consecutive stops as 5 kms (for example, distance between A and C is 10 kms). Both onward and backward journeys are possible (i.e., travelling from E to B is also possible).

Get source and destination from the user and calculate the distance to be travelled. Source and destination can take any of the values from the array 'stops' mentioned above. Get the number of tickets also as input.

Now according to the distance and no of tickets, fare has to be calculated as follows:

Distance ( in kms)	Rate per ticket(in Rs)
0 to 5	10
6 to 10	20
11 to 15	30
16 to 20	40
above 21	50

If the distance to be travelled is more than 10 kms, and the number of tickets purchased is more than 5, then apply a 10% discount to the total fare and display the final amount to be paid by the user. Use functions wherever applicable.

**Note:** A good modular program with multiple functions defined for various checks and calculations will fetch full marks for this question. If no functions are used and you compute everything only inside the `main()` function, then such codes will only get partial marks.

---