COMSATS University Islamabad

Lahore Campus

**Department of Computer Science**

**Sessional 2 BCS – Spring 2021**

**Course Title:** Programming Fundamentals  **Course Code:**

**Resource Person:** Dr. Muhammad Aksam Iftikhar **Credit Hours:** 3

**Time allowed:** 1.5 Hours **Total Marks:** 20

**Note:** All questions carry 1 mark, unless otherwise stated.

**Q1. [MCQs: 3 Marks]**

1. [2 Marks] Which of the following are incorrect array declarations? Select All that apply.
   1. *int c[3] = {1, 2, 3, 4};*
   2. int c[] = {1, 2, 3, 4};
   3. int c[10] = {1, 2, 3, 4};
   4. int c[SIZE] = {1, 2, 3, 4}; // SIZE is a constant equal to 4
   5. int c[];
2. Assume char cArr[] = {‘X’, ‘Y’, ‘Z’, ‘A’, ‘B’, ‘C’}; int iArr[] = {1, 2, 3, 0, 4, 5};

What will be output by the following statement: cout<<cArr[ iArr[3] + 1 ];

1. X
2. Y
3. Z
4. A
5. Error in the statement

**Q2. [Short Questions (programming): 5 Marks]**

1. [3 Marks] Assume int iArr1[5] = {2, 3, 0, 4, 5}, iArr2[5] = {3, 4, 5, 1, 3};

Write down the error (if any) with each of the following statements.

1. iArr1 = iArr2;
2. cout<<iArr[];
3. iArr1[5] = 1;
4. Write down a SINGLE C++ statement which declares an array of 5 integers initialized to 0, an uninitialized integer variable x and another integer variable y initialized to -10.
5. Write down a function prototype, which receives 2 double parameters by reference and 2 integer parameter by value and returns nothing.

**Q3. [Short Questions (descriptive): 5 Marks]**

1. [3 Marks] Briefly describe why we should avoid global variables in general (unless necessary)? Rather than passing parameters to functions, isn’t it a good idea to declare all variables global so that they are directly accessible by functions?
2. [2 Marks] Briefly describe the difference between the break and continue statements.

**Q4. [Program development: 7 Marks]**

1. In this program, you have to write a function called addTime, which receives 3 parameters. The first 2 parameters are hours and minutes (passed by reference), while the 3rd parameter is the number of minutes to add to this time (passed by value). For example, if the values of first 2 parameters are 10 and 20, while the 3rd parameter is 50. Then, the function should compute and return the new time as 11:10 (by adding 50 minutes to 10:20), i.e. hours value should be updated to 11 and minutes value to 10. (**Hint:** you might use the modulus operator to control the overflow of hours and minutes)

Inside the main function: Test this function by calling it 3 times for different user input values of hours/minutes and minutes to add (see the sample output). Note that the hours and minutes inputs should be validated. The updated time should be shown after each user input.

Text

Description automatically generated