Practice questions v8: Functions and local/global variables (Solution)

MCQs

1. Which of the following gives the memory address of the first element in array?
   A) array[0];
   B) array[1];
   C) &array[0];
   D) array;
   E) Both C and D

   Answer: E
   Explanation: Array names acts as a base address. Moreover if we place ampersand sign then it will also give address.

2. Which of the following accesses the seventh element stored in array?
   A) array[6];
   B) array[7];
   C) array(7);
   D) array;

   Answer: A
   Explanation: index starts from 0, therefore 7th element will have an index 6.

3. What is the scope of the variable declared in the user defined function?
   A. Whole program
   B. Only inside the {} block
   C. The main function
   D. None of the above
   Ans : B
   Explanation: The variable is valid only in the function block as in other.

4. How are many minimum numbers of functions need to be presented in c++?
   A. 0
   B. 1
   C. 2
   D. 3
   Ans : B
   Explanation: The main function is the mandatory part, it is needed for the execution of the program to start.

5. What are mandatory parts in function declaration?
   A. return type,function name
   B. return type,function name,parameters
   C. both a and b
   D. none of the mentioned

Ans: A
Explanation: In a function, return type and function name are mandatory all else are just used as a choice. We just have to mention data type of parameters not parameters themselves.

6. which of the following is used to terminate the function declaration?
   A. :
   B.)
   C. ;
   D. none of the mentioned
   Ans: C

7. Which is more effective while calling the functions?
   A. call by value
   B. call by reference
   C. none of the mentioned

   Ans: B
   Explanation: This is because call by value creates a memory (local variables in parameters list) and copies the values passed in function call to these local variables. This utilizes extra memory. Call by reference on the other hand does not allocates new memory, instead just create a reference to the variables that are passed in function call. This saves memory.

8. What will you use if you are not intended to get a return value?
   A. static
   B. const
   C. volatile
   D. void

   Answer: D
   Explanation: Void is used to not to return anything.

9. Where does the return statement returns the execution of the program?
   A. main function
   B. caller function
   C. same function
   D. none of the mentioned

   Answer: B
   Explanation: caller function

10. What are the advantages of passing arguments by reference?
    A. Changes to parameter values within the function also affect the original arguments.
    B. There is need to copy parameter values (i.e. less memory used)
    C. All of the mentioned

    Answer: C

Explanation: All of the mentioned

Identify the scope

Consider the following code. Which of the following are global variables (variables with a global scope)?

A. `hours_worked, minutes_worked`
B. `RATE`
C. `hours_worked, minutes_worked, RATE`
D. hours, minutes, bill

```
1)      #include <iostream>
2)      using namespace std;
3)      const double RATE = 150.00;
4)      double fee(int hours_worked, int minutes_worked);
5)      int main()
6)      {
7)          int hours, minutes;
8)          double bill;
9)          cout << "Welcome \n";
10)         cin >> hours >> minutes;
11)         bill = fee(hours, minutes);
12)         cout << "For " << hours << " hours and " << minutes
13)         << " minutes, your bill is $" << bill << endl;
14)         return 0;
15)     }
16)     double fee(int hours_worked, int minutes_worked)
17)     {
18)        int quarter_hours;
19)        minutes_worked = hours_worked * 60 + minutes_worked;
20)        quarter_hours = minutes_worked / 15;
21)        return (quarter_hours * RATE);

22)     }
```

Consider the code above. What will be the scope of variables involved?

Variable	Scope (from line#)	Scope (to line#)
RATE		
hours_worked		
minutes_worked		
hours		
minutes		
bill		
quarter_hours		

Variable	Scope (from line#)	Scope (to line#)
RATE	3	22
hours_worked	16	22
minutes_worked	16	22
hours	7	15
minutes	7	15
bill	8	15
quarter_hours	18	22

Output questions

To solve the following pls don't refer to compiler. Try it by dry running the code on a piece of paper. Only then this will help! For normal practice, you can attempt **only first three** questions and skip the rest.

1. Consider the following statements:

```
double num1, num2, num3;
int int1, int2, int3;
int value;
num1 = 5.0; num2 = 6.0; num3 = 3.0;
int1 = 4; int2 = 7; int3 = 8;
```

   and the function prototype:

```
double cube(double a, double b, double c);
```

   Which of the following statements are valid? If they are invalid, explain why.

   a. value = cube (num1, 15.0, num3);
   Valid
   b. cout << cube(num1, num3, num2) << endl;
   Valid
   c. cout << cube(6.0, 8.0, 10.5) << endl;
   Valid
   d. cout << cube(num1, num3) << endl;
   Invalid
   e. value = cube(num1, int2, num3);
   Valid
   f. value = cube(7, 8, 9);
   Valid
   g. value = cube(int1, int2, int3);
   Valid

Solution: All are valid except (d) because the count of parameters in the function header does not match with the function call

2. What is the output of the following program?

```
#include <iostream>
using namespace std;
void tryMe(int& v);
int main()
{
        int x = 8;
        for (int count = 1; count < 5; count++)
        tryMe(x);
        return 0;
}
void tryMe(int& v)
{
        static int num = 2;
        if (v % 2 == 0)
        {
                num++;
                v = v + 3;
        }
        else
        {
                num--;
                v = v + 5;
        }
cout << v << ", " << num << endl;
}
```

Solution:

```
11, 3
16, 2
19, 3
24, 2
```

3. Consider the following functions:

```
int secret(int x)
{
        int i, j;
        i = 2 * x;
        if (i > 10)
                j = x / 2;
        else
                j = x / 3;
        return j - 1;
}
int another(int a, int b)
{
        int i, j;
        j = 0;
        for (i = a; i <= b; i++)
                j = j + i;
        return j;
}
```

What is the output of each of the following program segments? Assume that x, y, and k are int variables.

a.
```
x = 10;
cout << secret(x) << endl;
```
Output: 4

b.
```
x = 5; y = 8;
cout << another(x, y) << endl;
```
Output: 26

c.
```
x = 10; k = secret(x);
cout << x << " " << k << " " << another(x, k) << endl;
```
Output: 10 4 0

d.
```
x = 5; y = 8;
cout << another(y, x) << endl;
```
Output: 0

4. Consider the following function:

```
int mystery(int x, double y, char ch)
{
        int u;
        if ('A' <= ch && ch <= 'R')
                return(2 * x + static_cast<int>(y));
        else
                return(static_cast<int>(2 * y) - x);
}
```

What is the output of the following C++ statements?

a) `cout << mystery(5, 4.3, 'B') << endl;`
   Output: 14
b) `cout << mystery(4, 9.7, 'v') << endl;`

Output: 15

c) cout << 2 * mystery(6, 3.9, 'D') << endl;

Output: 30

5. Show the output of the following program:

```
#include <iostream>
using namespace std;
int mystery(int);
int main()
{
        int n;
        for (n = 1; n <= 5; n++)
        cout << mystery(n) << endl;
        return 0;
}
int mystery(int k)
{
        int x, y;
        y = k;
        for (x = 1; x <= (k - 1); x++)
        y = y * (k - x);
        return y;
}
```

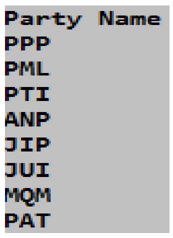
Output:

```
1
2
6
24
120
```

Programming challenges

For normal practice, you can attempt **only first three** challenges and skip the rest.

1. Election happened in a country with 8 parties contesting the election. Make a program to check who won the general election. Take the votes for 8 parties as an input from the user in an array, in main() function. Implement a function **checkWinner** that takes votes an argument/ parameter and report the winner party on the console. Also implement a function **FindHistogram** that takes prints the frequency of votes for each party using asterix: e.g. if PTI has 10 votes and PMLN has 4 votes then the output should be like:
   PMLN ****
   PTI **********



```
Party Name
PPP
PML
PTI
ANP
JIP
JUI
MQM
PAT
```

2. Implement a function FindDistinctArrayElements that declares an array of 5 elements. The function should then find the distinct elements and print them on the console. A distinct element of an array is the one which has occurred only once in an array.
3. Implement a function **ExtractEven** that takes an array as an argument. The function should then create an array containing only elements at even indexes. The function then calls another function **printEvenArray** and pass this newly created array as an argument while calling.
4. Implement a function **Replace** that takes an array as an argument. The function should then input two things from the user: **index** and **value**. It should then replace the value at **index** of an array with **value.**
5. Implement a function **Sum2D** that takes three parameters: array, index1 and index2. Index1 is the row index and index2 is the column index for the row/column whose sum must be found. The function should then report both the row sum and col sum.
6. Write a program which ask user to input a string. You should implement following functions on give string.

1.Count(string): count number of letters in the string and print them on console.
2. Find(char letter): Ask user to input a letter and find the letter in the string.
3. CompareStrings(string): Ask user to enter another string and compare both string either equal or not (w.r.t length). Display the answer on the console.
4. AppendingString: Ask user to enter string to append it to the previous string.
5. Replace(): Ask user to enter 2 values:

- Index of the string, user want to replace.
- Value to be replaced on the specified index.

Note: you can use built in function for all functions.
6. Write a program that uses the function isNumPalindrome that finds if a given number is a palindrome or not. Palindrome is a number which is the same as its reverse. E.g. 12321. You might need to use modulus and division operator to implement that logic. Refer to our previous lectures in which we discussed the role of modulus and division operators.