

Functions

A. Multiple-choice questions (5.0 marks)

A.1. Which is NOT a proper prototype?

Answer:

B. double func(char b)

Explain:

Missing the semicolon “;”.

A.2. Assume that the function `func` has already been declared as follows.

```
int func(char &a, char &b);
```

Which of the following is a valid function call with parameters `x` and `y` of `char` type?

Answer:

D. `func(x, y);`

Explain:

According to <http://www.cplusplus.com/doc/tutorial/functions/>, we use `function-name(parameter1, parameter2, ...)` to pass *parameter1*, *parameter2*, ... into parameters of the functions (Call by Value).

A.3. What is the result of the following program?

```
int change(int &a){  
    a = 10;  
    return a;  
};  
void main() {  
    int a;  
    change(a);  
    cout << a;  
}
```

Answer:

A. 10

Explain:

Biến `a` được khai biến là kiểu số nguyên, sau đó `a` được truyền tham chiếu vào hàm **change**. Qua keyword “&” vì vậy `a` trong hàm `change` là `a` bên ngoài, đồng nghĩa với việc thay đổi `a` trong hàm `change` là thay đổi `a` bên ngoài (hàm `main`).

A.4. What is the output of the `Boys` variable on the marked lines?

```
#include <iostream>  
using namespace std;  
void F1(int b, int g) {
```

```

        b += 3, g += 4;
        cout << "\nF1";
        cout << "\n\tBoys = " << b; // #1
        cout << "\n\tGirls = " << g;
    }
    void F2(int &b, int &g) {
        b = b + 8, g = g + 5;
        cout << "\nF2";
        cout << "\n\tBoys = " << b; // #3
        cout << "\n\tGirls = " << g;

    int main() {
        int Boys = 3, Girls = 5;
        F1(Boys, Girls);
        cout << "\nAfter calling F1, within main()";
        cout << "\n\tBoys = " << Boys; // #2 cout <<
        "\n\tGirls = " << Girls;

        F2(Boys, Girls);
        cout << "\nAfter calling F2, within main()";
        cout << "\n\tBoys = " << Boys; // #4
        cout << "\n\tGirls = " << Girls;
    }
}

```

Answer:

A. #1: 6 #2: 3
 #3: 11 #4: 11

Explain:

Tại #1, Màn hình in ra 6 vì nhận giá trị của Boys là 3, sau đó tạo một biến copy của Boys, rồi cộng thêm 3 cho biến đó. In ra 6.

Tại #2, Vì sau khi F1 thực thi, giá trị của Boys không thay đổi nên in 3

Tại #3, Trong F2, giá trị được truyền là tham chiếu. Giá trị của b được xem như là của Boys, cộng cho giá trị đó thêm 8. Vậy ta có $3 + 8 = 11$, in 11. Sau hàm F2, giá trị của Boys thay đổi thành giá trị của b.

Tại #4, Vì sau F2 thực thi, giá trị của Boys thay đổi nên in ra 11.

Vậy đáp án A đúng.

A.5. If we move the line “int Boys = 3, Girls = 5;” from main() and place it in the global scope, what will happen?

Answer:

A. The output would be the same.

Explain:

Vì Boys và Girls không xuất hiện trong 2 hàm F1 và F2 (và không chịu tác động bởi 2 hàm trên) mà chỉ xuất hiện trong hàm main nên output không thay đổi.

B. Essay questions (5.0 marks)

B.1. Which statement is used to stop a function immediately? Give an example of such statement.

Statement được dùng để ngưng hàm ngay lập tức là **return**.

Answer:

```
return
```

Example:

```
int calculate(int a) {  
    a = a + 1;  
    return a; // vì tới đây chương trình bị ngắt, nên các dòng dưới không được chạy  
    a = a + 2;  
}  
  
int main() {  
    cout << calculate(2); // Expected Output: 3  
}
```

B.2. A function, by default, can only return one value of the return type. Can you make it to return more values? Give an example to demonstrate how to do that.

Like this:

```
#include <iostream>  
using namespace std;  
  
void change(int &a, int &b, int &c) {  
    a = 1;  
    b = 2;  
    c = 3;  
}  
  
int main() {  
    int a, b, c;  
    change(a, b, c);  
}
```

B.3. Is it possible to declare two variables of the same name `myvar` as shown below? If yes, what will be printed out? If no, explain.

```
int myvar = 10;  
void somefunc() {  
    int myvar;  
    myvar = 20;  
    cout << myvar << endl;  
}  
  
int main()  
{  
    somefunc();  
    cout << myvar << endl;  
}
```

Answer/Explain:

Yes, it will print 20, because `cout << myvar;` will output the local myvar value.

If it wants to print 10, it could be `cout << ::myvar;`

B.4. Are the results of the two following programs the same? If no, what makes the difference?

<pre>void mul (int n){ n = 20; } void main (){ int n = 4; mul(n); cout << n; }</pre>	<pre>void mul(int &n){ n = 20; } void main (){ int n = 4; mul(n); cout << n; }</pre>
--	--

Answer:

Các kết quả không giống nhau, như bên trái là truyền tham trị (pass by value) trong khi bên phải là truyền tham chiếu (pass by reference), vậy nên bên trái sau khi hàm mul thực hiện giá trị n của main() không đổi, trong khi bên phải giá trị n thay đổi.

Output bên trái là 4, output bên phải là 20.

B.5. Consider the following three function prototypes. Which of them are defined properly? Point out the error(s) in the improper function prototypes.

1. `int Sum(int a, int b = 2, int c = 3);`
2. `int Sum(int a = 5, int b);`
3. `int Sum(int a = 0, int b, int c = 3);`

Không sai gì hết

C. Programming questions (10.0 marks)

Students do not have to check the validity of the input.

- C.1.** Write a function that takes an integer n and outputs the n^{th} element of a Fibonacci sequence.
Write a program that prompts the user to enter a positive integer N ($N \geq 3$) and then iteratively calls the function to output N first elements of a Fibonacci sequence.
- C.2.** Write a function that takes an integer and prints out the multiplication between this number and numbers from 1 to 10. Write a program that calls the function to produce a multiplication table.
- C.3.** Write a function that takes an integer and outputs the square of that integer. Write a program that prompts the user to enter a positive integer N, and then uses the function to calculate the sum of $1^2 + 2^2 + \dots + (N-1)^2 + N^2$.
- C.4.** Write a function that takes two positive integers and outputs their greatest common divisor. The algorithm should be naively based on loops. Write a program that calculate the least common multiple of two positive integers, given their greatest common divisor.
- C.5.** Repeat question C.4. but this time, use the Euclidean algorithm to determine the greatest common divisor.