EE2310 C++程式設計 HW3 (Ch 6)

**Part 1. Choice選擇題: 26%**

1) Breaking a program up into a set of manageable sized functions is called \_\_\_\_\_\_\_\_ programming.

A) functional

B) modular

C) break up

D) high-level

E) low-level

2) A function other than the main function is executed \_\_\_\_\_\_\_\_.

A) when it is first defined

B) only once

C) whenever it is called

D) when the main function finishes executing

E) never

3) In a function header, in addition to the name of the function, you are required to furnish \_\_\_\_\_\_\_\_.

A) a data type for each parameter

B) an identifier name for each parameter

C) the data type of the return value

D) All of the above

E) B and C, but not A.

4) In a function call, in addition to the name of the function, you are required to furnish \_\_\_\_\_\_\_\_.

A) a data type for each argument

B) an identifier name or constant for each argument

C) the data type of the return value

D) All of the above

E) A, and B, but not C.

5) Functions are ideal for use in menu-driven programs. When a user selects a menu item, the program can \_\_\_\_\_\_\_\_ an appropriate function to carry out the user's choice.

A) call

B) return

C) define

D) declare

E) randomly select

6) The value in \_\_\_\_\_\_\_\_ local variable is retained between function calls.

A) a global

B) an internal

C) a static

D) a dynamic

E) no

7) When used as a parameter, a \_\_\_\_\_\_\_\_ variable allows a function to access and modify the original argument passed to it.

A) static

B) value

C) reference

D) floating-point

E) default value

8) Two or more functions may have the same name provided that \_\_\_\_\_\_\_\_.

A) they do different things

B) one has a prototype and the other doesn't

C) their parameter lists are different

D) their return types are different

E) either C or D is true.

9) The \_\_\_\_\_\_\_\_ statement causes a function to end and the flow of control to move back to the point where the function call was made.

A) end

B) break

C) continue

D) return

E) exit

10) The \_\_\_\_\_\_\_\_ function causes the entire program to terminate, regardless of which function or control mechanism is executing.

A) terminate()

B) return()

C) continue()

D) exit()

E) break()

11) A function \_\_\_\_\_\_\_\_ is a statement that causes a function to execute.

A) prototype

B) header

C) definition

D) call

E) parameter list

12) A void function is one that \_\_\_\_\_\_\_\_.

A) has an empty function body

B) is never called

C) returns no value

D) has no parameters

E) returns a zero

13) A function can have \_\_\_\_\_\_\_\_ parameters, and it can have either zero or one return value(s).

A) zero to many

B) either zero or one

C) either one or two

D) a maximum of ten

E) no

14) When only a copy of an argument is passed to a function, it is said to be passed \_\_\_\_\_\_\_\_.

A) by copy

B) by reference

C) informally

D) by value

E) by default value

15) A function \_\_\_\_\_\_\_\_ eliminates the need to place the function definition before all calls to the function.

A) header

B) prototype

C) argument

D) parameter

E) that is void

16) In the following statement, what is 22.0?

cout<<sqrt(22.0);

A) a memory location

B) a parameter

C) an argument

D) a default value

E) an lvalue

17) In C++ numeric global variables are \_\_\_\_\_\_\_\_ by default and numeric local variables are \_\_\_\_\_\_\_\_ by default.

A) initialized to zero, initialized to zero

B) not initialized, not initialized

C) not initialized, initialized to zero

D) initialized to zero, not initialized

E) None of the above

18) A(n) \_\_\_\_\_\_\_\_ argument is one that is automatically passed to a parameter when the argument is left out of the function call.

A) floating-point

B) actual

C) null

D) default

E) static

19) A function other than the main function is executed \_\_\_\_\_\_\_\_.

A) when it is first defined

B) only once

C) whenever it is called

D) when the main function finishes executing

E) never

20) In a function prototype, in addition to the name of the function, you are required to furnish \_\_\_\_\_\_\_\_.

A) a data type for each parameter

B) an identifier name for each parameter

C) the data type of the return value

D) All of the above

E) A and C, but not B.

21) Functions are ideal for use in menu-driven programs. When a user selects a menu item, the program can \_\_\_\_\_\_\_\_ an appropriate function to carry out the user's choice.

A) call

B) return

C) define

D) declare

E) randomly select

22) In the following program, what would be the output?

void f(int x, int &y, int& z);

int main(){

int a=10, b=20, c=30;

f(a,b,c);

cout << a << “ “ << b <<” “ << c << endl;

return 0;

}

void f(int x, int &y, int& z){

x += 1; y += 2; z += 3;

cout << x << “ “ << y << “ “ << z << “, “;

}

A) 10 20 30, 10 20 30

B) 1 2 3, 10 2 3

C) 11 22 33, 10 20 30

D) 11 22 33, 11 22 33

E) 11 22 33, 10 22 33

23) A \_\_\_\_\_\_\_\_ is a dummy function that is called instead of the actual function it represents, to test that the call to and return from the function are working correctly.

A) stub

B) driver

C) test function

D) void function

E) prototype function

24) A \_\_\_\_\_\_\_\_ is a program module whose purpose is to test other modules by calling them.

A) stub

B) driver

C) main function

D) dummy program

E) pseudocode routine

25) Which of the following functions will be used in the function call: f(98, 99); ?

A) void f(int n, double m);

B) void f(double n, int m);

C) void f(double n, double m);

D) void f(int n, int m);

E) none of the above

26) When should pass-by-reference be used?

A) when data values being input in a function need to be known by the calling function

B) when a function must change existing values in the calling function

C) when a file stream object is passed to a function

D) when a function needs to have more than one return values.

E) all of the above

**Part 2. True/False是非題: 14%**

**T 1.** It is possible for a function to have some parameters with default arguments and some without.

**F 2.** If the closing brace of a function body is reached, the flow of control moves to the next function in the file.

**T 3.** One reason for using functions is to break programs into a set of manageable units, or modules.

**F 4.** A function with a return type of bool must return a value of either correct or incorrect.

**F 5.** When you make a function call, the order of the arguments you send does not matter as long as the number of arguments matches the number of parameters the function has.

**F 6.** Due to the static storage duration and file scope properties of global variables, it is considered good programming practice to suggest your use of them.

**F 7.** Both function headers and function calls must list the data types of all data being passed to the function.

**F 8.** You may use the exit() function to return the flow of control from a function back to main(), regardless of where the function was called from.

**F 9.** A local variable defined in a function can only be directly accessed in that function, i.e., it has automatic storage duration and block scope properties. However, if that local variable is specified with static, then it can be directly accessed outside the function.

**T 10.** When a copy of an argument cannot reasonably or correctly be made, such as when the argument is a file stream object, it must be passed by reference.

**T 11.** The following code segment illustrates a valid case of default parameter:

int f(int a, int b, double c); //function prototype

…

int f(int a, int b=1, double c = 0.5){….. } //function definition

int main(){…}

**F 12.** One can always tell whether a function parameter is pass-by-value or pass-by-reference by observing only the function call itself.

**T 13.** When a function has a mixture of parameters both with and without default arguments, the parameters with default arguments must be defined last.

**T 14.** If we want to use the function shown below to inverse the value 20.0, we can just call the function: inv(20.0);

void inv(double& x){ if(x!=0) x = 1/x;}