1. Hãy viết một hàm majority() lấy ba tham số kiểu bool, và trả về true nếu có ít nhất hai tham số có giá trị true, nếu không thì trả về false. Không được dùng lệnh if trong hàm.

2.            Hãy viết một hàm eq() lấy ba tham số: hai mảng int một chiều và một số nguyên n. Hàm trả về true nếu n phần tử của hai mảng bằng nhau đôi một, nếu không thì trả về false.

Eq(int n, int a[n], int b[n])

{

Int t=0;

            For(int i=0;i<n;i++)

            {

                        If(a[i]==b[i]) t++;

            }

If(t==n) return true;

Else return false;

}

3.            Cho hàm cube sau:

           void cube(int i) {  
              i = i \* i \* i;  
           }

            Hỏi trong vòng lặp sau, hàm cube() chạy bao nhiêu lần?

for (int i = 0; i < 1000; i++)  
        cube(i);

1000 lần

4.            Chương trình sau đây cho kết quả gì?

void negate(int a) {  
    a = -a;  
}  
  
int main() {  
    int a = 17;  
    cout << a;  
    negate(a);  
    cout << a;  
}

17 17

5.            Hãy lập một bộ test để thử xem trong các hàm dưới đây, đâu là cài đặt đúng cho một hàm min trả về số nhỏ nhất trong 04 tham số. Có thể hardcode giá trị các test thẳng vào chương trình test.

int min(int a, int b, int c, int d) {  
   // if a is the smallest return it  
   if (a <= b && a <= c && a <= d) return a;  
  
   // otherwise, if b is the smallest of b, c, and d, return it  
   if (b <= c && b <= d) return b;  
  
   // otherwise, return the smaller of c and d  
   if (c <= d) return c;  
   return d;  
}  
  
int min(int a, int b, int c, int d) {  
   int min = a;  
   if (b < min) min = b;  
   if (c < min) min = c;  
   if (d < min) min = d;  
   return min;  
}  
  
int min(int a, int b, int c, int d) {  
   if (a < b && a < c && a < d) return a;  
   if (b < c && b < d)          return b;  
   if (c < d)                   return c;  
   return d;  
}  
  
int min(int a, int b, int c, int d) {  
   if (a <= b) {  
      if (a <= c) {  
         if (a <= d)  return a;  
         else return d;  
      }  
      if (c <= d) return c;  
      else return d;  
   }  
   if (b <= c) {  
      if (b <= d) return b;  
      else return d;  
   }  
   else if (c <= d) return c;  
   return d;  
}  
  
int min(int a, int b) {  
   if (a <= b) return a;  
   else        return b;  
}  
int min(int a, int b, int c, int d) {  
   return min(min(a, b), min(c, d));  
}

6.            Cho hai hàm overload sau:

void f(int x, double y) {

   cout << "f(int, double)";

}

void f(double x, int y) {

   cout << "f(double, int)";

}

Lời gọi hàm này sẽ cho kết quả gì?

f(1,2);

7.            Tác dụng của lệnh return là gì?

A. Dùng khi muốn thoát khỏi hàm  
B. Trả lại kết quả cho hàm  
C. Bắt buộc phải có trong hàm  
D. Cả A và B

8.            Có thể truyền vào bao nhiêu đối số trong một hàm?  
A. Bao nhiêu cũng được.  
B. Ít nhất là 1.  
C. Nhiều nhất là 10.  
D. Chỉ duy nhất 1.

9.            Khai báo prototype nào sau đây là không hợp lệ?  
A. int funct(char x, char y);  
B. double funct(char x)  
C. void funct();  
D. char x();

10.          Kiểu trả về của hàm với prototype: "int func(char x, float v, double t);" là:  
A. char  
B. int  
C. float  
D. double

11.          Lời gọi hàm nào sau đây là hợp lệ (giả sử hàm đấy tồn tại)?  
A. funct;  
B. funct x, y;  
C. funct();  
D. int funct();

12.          Đâu là định nghĩa một hàm hoàn chỉnh?  
A. int funct();  
B. int funct(int x) {return x=x+1;}  
C. void funct(int) {cout<<"Hello"}  
D. void funct(int) {cout<<"Hello"}

13.          Giả sử một hàm tên là function1 có một biến tên là sam được khai báo bên trong định nghĩa hàm function1, và một hàm tên là function2 cũng có một biến tên là sam được khai báo bên trong định nghĩa của function2. Điều gì sẽ xảy ra khi chạy chương trình (giả thiết mọi thứ khác đều ổn):  
A. Không biên dịch được;  
B. Biên dịch được nhưng không chạy được;  
C. Chạy được nhưng sẽ gặp lỗi logic không mong muốn;  
D. Chạy đúng như mong muốn;

14.          Cho biết kết quả của đoạn code sau:  
#include <iostream>  
void display (int k)  
{  
  int j;  
  std::cout << “Hello” << std::endl;  
}  
int main(int argc, char\*\* argv)  
{  
  int s;  
  display(s);  
  
  return 0;  
}

A. Chương trình gặp lỗi biên dịch vì 2 biến s, k không giống nhau  
B. Chương trình gặp lỗi biên dịch vì 2 biến s, k không được khởi tạo trước khi truyền cho hàm  
C. Chương trình biên dịch thành công nhưng gặp lỗi khi chạy vì 2 biến s, k không giống nhau  
D. Chương trình biên dịch thành công và kết quả khi chạy là : Hello

15.          Đoạn code sau in ra kết quả như thế nào?  
#include <iostream>  
int foo(int y);  
int foo(int x)  
{  
  return x+1;  
}  
int main(int argc, char\*\* argv)  
{  
  int x = 3;  
  int y = 6;  
  
  std::cout << foo(x) << std::endl;  
  
  return 0;  
}  
A. 3  
B. 4  
C. 9  
D. Lỗi

16.          Hàm inline được định nghĩa trong  
A. Thời gian chạy  
B. Thời gian biên dịch  
C. Tùy vào nó được gọi khi nào  
D. Cả B và C

17.          Nếu một hàm được viết để sử dụng trong một chương trình sẵn có thì nó có thể được đưa vào thư viện hàm để các chương trình khác có thể sử dụng được  
A. Đúng  
B. Sai

18.          Số tham số và thứ tự các tham số thật phải đúng như các tham số hình thức trong khai báo hàm  
A. Đúng  
B. Sai

19.          Không cần chỉ định kiểu trả về của hàm khi hàm đó trả về kiểu dữ liệu nào sau đây:  
A. int  
B. void  
C. float  
D. char

20.          Định nghĩa hàm nào sau đây không hợp lệ  
A. void foo(){}  
B. void foo(void){}  
C. foo(void){}  
D. void foo{}

21.          Khi gọi hàm, các tham số được truyền vào phải giống các tham số hình thức ở  
A. Kiểu trả về  
B. Kiểu dữ liệu  
C. Tên  
D. Giá trị

22.          Nếu một hàm được gọi trước khi nó được định nghĩa thì điều kiện là gì?  
A. Kiểu trả về của hàm phải là kiểu void  
B. Kiểu đầu vào của hàm phải là kiểu void  
C. Trước khi gọi hàm nó phải được khai báo  
D. Không thể được.

23.          Hàm main() là một ví dụ về  
A. Hàm sơ cấp  
B. Hàm kiểu void  
C. Hàm do người dùng định nghĩa  
D. Hàm thư viện (pre-defined)

24.          Một hàm không trả về giá trị thì không thể có mặt trong biểu thức  
A. Đúng  
B. Sai

25.          Phát biểu nào sau đây là không đúng?  
A. Hàm có thể được định nghĩa trong hàm main()  
B. Hàm có thể được khai báo trong hàm main()  
C. Hàm có thể được gọi trước khi nó được định nghĩa  
D. Hàm chỉ có thể được định nghĩa bên ngoài hàm main()

26.          Phát biểu nào sau đây về hàm main() là đúng  
A. Là hàm được khai báo đầu tiên khi thực hiện chương trình  
B. Là hàm luôn được gọi đầu tiên khi chương trình thực hiện  
C. Là hàm không được phép truyền tham số vào  
D. Là hàm có thể gọi bình thường như các hàm người dùng định nghĩa khác.