

**NGÂN HÀNG CÂU HỎI ÔN TẬP KIỂM TRA GIỮA KÌ, HK1 NH 2022-2023**  
**MÔN: TIN HỌC LỚP 11**

**Câu 01.** Chương trình dịch là:

- A. Chương trình có chức năng chuyển đổi chương trình được viết bằng ngôn ngữ lập trình bậc cao thành chương trình thực hiện được trên máy tính cụ thể.
- B. Chương trình có chức năng chuyển đổi chương trình được viết bằng ngôn ngữ lập trình bậc thấp thành ngôn ngữ bậc cao.
- C. Chương trình có chức năng chuyển đổi chương trình được viết bằng ngôn ngữ ngôn ngữ máy sang ngôn ngữ lập trình cụ thể.
- D. Chương trình có chức năng chuyển đổi chương trình được viết bằng ngôn ngữ máy sang hợp ngữ.

**Câu 02.** Biên dịch là:

- A. Chương trình dịch, dịch toàn bộ chương trình nguồn thành một chương trình đích có thể thực hiện trên máy, không thể lưu trữ để sử dụng lại khi cần thiết.
- B. Chương trình dịch, dịch toàn bộ chương trình nguồn thành một chương trình đích có thể thực hiện trên máy và có thể lưu trữ để sử dụng lại khi cần thiết.
- C. Chương trình dịch, dịch toàn bộ ngôn ngữ lập trình bậc thấp sang ngôn ngữ lập trình bậc cao.
- D. Chương trình dịch, lần lượt dịch và thực hiện từng câu lệnh.

**Câu 03.** Thông dịch là:

- A. Chương trình dịch, dịch toàn bộ chương trình nguồn thành một chương trình đích có thể thực hiện trên máy, không thể lưu trữ để sử dụng lại khi cần thiết.
- B. Chương trình dịch, dịch toàn bộ chương trình nguồn thành một chương trình đích có thể thực hiện trên máy và có thể lưu trữ để sử dụng lại khi cần thiết.
- C. Chương trình dịch, dịch toàn bộ ngôn ngữ lập trình bậc thấp sang ngôn ngữ lập trình bậc cao.
- D. Chương trình dịch, lần lượt dịch và thực hiện từng câu lệnh.

**Câu 04.** Sự giống nhau giữa thông dịch và biên dịch là:

- A. Không phải chương trình dịch.
- B. Đều là chương trình dịch.
- C. Đều dịch từ ngôn ngữ lập trình bậc thấp sang ngôn ngữ lập trình bậc cao.
- D. Đều dịch từ ngôn ngữ máy sang hợp ngữ.

**Câu 05.** Sự khác nhau giữa thông dịch và biên dịch là:

- A. Thông dịch: lần lượt dịch và thực hiện từng câu lệnh. Biên dịch: dịch toàn bộ chương trình nguồn thành một chương trình đích có thể thực hiện trên máy và có thể lưu trữ để sử dụng lại khi cần thiết.
- B. Biên dịch: lần lượt dịch và thực hiện từng câu lệnh. Thông dịch: dịch toàn bộ chương trình nguồn thành một chương trình đích có thể thực hiện trên máy và có thể lưu trữ để sử dụng lại khi cần thiết.
- C. Biên dịch: dịch toàn bộ ngôn ngữ lập trình bậc cao sang ngôn ngữ lập trình bậc thấp. Thông dịch: dịch toàn bộ ngôn ngữ lập trình bậc thấp sang ngôn ngữ lập trình bậc cao.

**D.** Thông dịch: dịch toàn bộ ngôn ngữ lập trình bậc cao sang ngôn ngữ lập trình bậc thấp. Biên dịch: dịch toàn bộ ngôn ngữ lập trình bậc thấp sang ngôn ngữ lập trình bậc cao.

**Câu 06.** Lập trình là:

- A.** Mô tả dữ liệu và diễn đạt các thao tác của thuật toán.
- B.** Sử dụng cấu trúc dữ liệu để mô tả dữ liệu.
- C.** Sử dụng cấu trúc dữ liệu và các câu lệnh của ngôn ngữ lập trình cụ thể để mô tả dữ liệu và diễn đạt các thao tác của thuật toán.
- D.** Sử dụng cấu trúc dữ liệu để diễn đạt các thao tác của thuật toán.

**Câu 07.** Chương trình nguồn là:

- A.** Chương trình viết bằng mã nhị phân.
- B.** Chương trình viết bằng ngôn ngữ máy.
- C.** Chương trình viết bằng ngôn ngữ lập trình bậc thấp.
- D.** Chương trình viết bằng ngôn ngữ lập trình bậc cao.

**Câu 08.** Chương trình đích là:

- A.** Chương trình viết bằng hợp ngữ.
- B.** Chương trình viết bằng ngôn ngữ máy.
- C.** Chương trình viết bằng ngôn ngữ lập trình C++.
- D.** Chương trình viết bằng ngôn ngữ lập trình bậc cao.

**Câu 09.** Phát biểu nào sau đây là đúng?

- A.** Chương trình viết bằng ngôn ngữ lập trình bậc cao nói chung không phụ thuộc vào loại máy.
- B.** Chương trình viết bằng ngôn ngữ lập trình bậc cao nói chung phụ thuộc vào loại máy.
- C.** Chương trình viết bằng ngôn ngữ lập trình bậc cao nói chung phụ thuộc vào rất nhiều yếu tố.
- D.** Chương trình viết bằng ngôn ngữ lập trình bậc thấp nói chung không phụ thuộc vào loại máy.

**Câu 10.** Phát biểu nào sau đây là sai?

- A.** Chương trình viết bằng ngôn ngữ máy có thể được nạp trực tiếp vào bộ nhớ và thực hiện ngay, còn chương trình viết bằng ngôn ngữ lập trình bậc cao phải được chuyển đổi thành chương trình trên ngôn ngữ máy mới có thể thực hiện được.
- B.** Chương trình đặc biệt có chức năng chuyển đổi chương trình được viết bằng ngôn ngữ lập trình bậc cao thành chương trình thực hiện được trên máy tính cụ thể được gọi là chương trình dịch.
- C.** Lập trình là sử dụng cấu trúc dữ liệu và các câu lệnh của ngôn ngữ lập trình cụ thể để mô tả dữ liệu và diễn đạt các thao tác của thuật toán.
- D.** Chương trình dịch nhận đầu vào là chương trình viết bằng ngôn ngữ lập trình bậc thấp (chương trình nguồn), thực hiện chuyển đổi sang ngôn ngữ lập trình bậc cao (chương trình đích).

**Câu 11.** Phát biểu nào sau đây là sai?

- A.** Để giải bài toán bằng máy tính phải viết chương trình miêu tả thuật toán giải bài toán đó.
- B.** Mọi người sử dụng máy tính đều phải biết lập chương trình.

- C. Máy tính điện tử hoàn toàn có thể chạy các chương trình.
- D. Một bài toán có thể có nhiều thuật toán để giải.

**Câu 12.** Ngôn ngữ lập trình là ngôn ngữ:

- A. Có tên là “ngôn từ thuật toán” hay còn gọi là “ngôn từ lập trình bậc cao” gần với ngôn từ toán học được cho phép miêu tả cách xử lý yếu tố độc lập với máy tính.
- B. Diễn đạt thuật toán để hoàn toàn có thể giao cho máy tính thực thi.
- C. Dưới dạng nhị phân để máy tính hoàn toàn có thể triển khai trực tiếp.
- D. Cho phép bọc lộ những tài liệu trong bài toán mà những chương trình sẽ phải xử lí.

**Câu 13.** Ngôn ngữ lập trình bậc cao là ngôn ngữ

- A. Có thể diễn đạt được mọi thuật toán.
- B. Mà máy tính không hiểu trực tiếp được, chương trình viết trên ngôn từ bậc cao trước khi chạy phải dịch sang ngôn ngữ máy.
- C. Thể hiện thuật toán theo những quy ước nào đó không phụ thuộc vào những máy tính đơn cử.
- D. Sử dụng từ vựng và cú pháp của ngôn từ tự nhiên (tiếng Anh).

**Câu 14.** Hợp ngữ là ngôn ngữ:

- A. Không viết bằng mã nhị phân, được phong cách thiết kế cho một số ít loại máy hoàn toàn có thể chạy trực tiếp dưới dạng kí tự.
- B. Có những lệnh được viết bằng kí tự nhưng về cơ bản mỗi lệnh tương tự với một lệnh máy. Để chạy được cần dịch ra ngôn từ máy.
- C. Mà các lệnh không viết trực tiếp bằng mã nhị phân.
- D. Mà máy tính hoàn toàn có thể triển khai được trực tiếp không cần dịch.

**Câu 15.** Ngôn ngữ máy là:

- A. Các ngôn từ mà chương trình viết trên chúng sau khi dịch ra hệ nhị phân thì máy hoàn toàn có thể chạy được.
- B. Ngôn ngữ để viết những chương trình mà mỗi chương trình là một dãy lệnh máy trong hệ nhị phân.
- C. Bất cứ ngôn từ lập trình nào mà hoàn toàn có thể diễn đạt thuật toán để giao cho máy tính thực thi.
- D. Diễn đạt thuật toán để hoàn toàn có thể giao cho máy tính triển khai.

**Câu 16.** Mỗi ngôn ngữ lập trình thường có các thành phần cơ bản là:

- A. Bảng chữ cái, cú pháp.
- B. Bảng chữ cái và ngữ nghĩa.
- C. Bảng chữ cái, cú pháp và ngữ nghĩa.
- D. Cú pháp và ngữ nghĩa.

**Câu 17.** Tên nào sau đây trong ngôn ngữ C++ là đặt đúng theo quy cách:

- A. \_Bai1
- B. Bai@1
- C. 1Bai 1
- D. Bai 1

**Câu 18.** Tên nào sau đây trong ngôn ngữ C++ là đặt sai theo quy cách:

- A. Bai#1

- B. Bai1
- C. \_Bai1
- D. Bai1\_

**Câu 19.** Chọn phát biểu đúng trong ngôn ngữ lập trình C++ ?

- A. Tên dành riêng là loại tên được ngôn ngữ lập trình quy định dùng với ý nghĩa xác định, người lập trình có thể được sử dụng với ý nghĩa khác.
- B. Tên do người lập trình đặt được dùng với ý nghĩa riêng, xác định bằng cách khai báo trước khi sử dụng. Các tên này được trùng với tên dành riêng.
- C. Tên dành riêng là loại tên được ngôn ngữ lập trình quy định dùng với ý nghĩa xác định, người lập trình không được sử dụng với ý nghĩa khác.
- D. Hằng là các đại lượng có giá trị thay đổi trong quá trình thực hiện chương trình.

**Câu 20.** Chọn phát biểu đúng khi nói về Hằng?

- A. Hằng là đại lượng thay đổi trong quá trình thực hiện chương trình.
- B. Hằng là các đại lượng có giá trị không thay đổi trong quá trình thực hiện chương trình. Bao gồm: Hằng số học, hằng logic, hằng xâu.
- C. Hằng là đại lượng bất kì.
- D. Hằng không bao gồm: số học và logic.

**Câu 21.** Chọn phát biểu đúng khi nói về biến ?

- A. Biến là đại lượng được đặt tên, dùng để lưu trữ giá trị và giá trị có thể được thay đổi trong quá trình thực hiện chương trình.
- B. Biến là đại lượng bất kì.
- C. Biến là đại lượng không thay đổi trong quá trình thực hiện chương trình.
- D. Biến là đại lượng được đặt tên, dùng để lưu trữ giá trị và giá trị không thay đổi trong quá trình thực hiện chương trình.

**Câu 22.** Hãy cho biết biểu diễn nào dưới đây không phải là biểu diễn hằng trong C++:

- A. 'A'
- B. 23
- C. TRUE
- D. 1.5

**Câu 23.** Chọn phát biểu đúng trong các phát biểu dưới đây:

- A. Bảng chữ cái của ngôn ngữ lập trình C++ không có dấu nháy kép ("").
- B. Ngoài bảng chữ cái, có thể dùng các kí tự thông dụng trong toán học để viết.
- C. Chương trình có lỗi cú pháp có thể được dịch ra ngôn ngữ máy nhưng không thực hiện được.
- D. Cú pháp là bộ quy tắc dùng để viết chương trình.

**Câu 24.** Đề chú thích 1 dòng trong C++ :

- A. Đặt dấu // ở đầu dòng cần chú thích.
- B. Đặt dòng cần chú thích trong cặp ngoặc {}.
- C. Đặt dấu @ ở đầu dòng cần chú thích.
- D. Đặt dấu # ở cuối dòng cần chú thích.

**Câu 25.** Phát biểu nào sau đây là đúng khi nói về tên dành riêng?

- A. Tên dành riêng do người lập trình đặt cần khai báo trước khi sử dụng.

- B. Là loại tên được ngôn ngữ lập trình quy định dùng với ý nghĩa xác định, người lập trình không được sử dụng với ý nghĩa khác.
- C. Là loại tên được ngôn ngữ lập trình quy định dùng với ý nghĩa xác định, người lập trình được phép sử dụng với ý nghĩa khác.
- D. Người lập trình khai báo tên dành riêng trước khi sử dụng.

**Câu 26.** Chọn phát biểu đúng khi chú thích nhiều dòng trong C++ :

- A. Đặt dấu // ở đầu dòng cần chú thích.
- B. Chú thích trên nhiều dòng được bắt đầu bằng ngoặc { và kết thúc bằng ngoặc }
- C. Đặt dấu @ ở đầu dòng cần chú thích.
- D. Chú thích trên nhiều dòng được bắt đầu bằng kí hiệu /\* và kết thúc bằng kí hiệu \*/

**Câu 27.** Phát biểu nào dưới đây là hợp lí nhất khi nói về hằng?

- A. Hằng là đại lượng nhận giá trị trước khi chương trình thực hiện.
- B. Hằng là đại lượng được đặt tên và có giá trị thay đổi trong quá trình thực hiện chương trình.
- C. Hằng có thể lưu trữ nhiều loại giá trị khác nhau.
- D. Hằng được chương trình dịch bỏ qua.

**Câu 28.** Chọn phát biểu đúng nhất khi nói về tên?

- A. Tên gọi là đại lượng nhận giá trị trước khi chương trình thực hiện.
- B. Tên gọi là đại lượng được đặt tên và có giá trị thay đổi trong quá trình thực hiện chương trình.
- C. Tên gọi có thể lưu trữ nhiều loại giá trị khác nhau.
- D. Tên gọi do người lập trình tự đặt theo quy tắc do từng ngôn ngữ lập trình xác định.

**Câu 29.** Hằng xâu trong C++ là:

- A. 'a'
- B. ab
- C. true
- D. "ab"

**Câu 30.** Hằng kí tự trong C++ là:

- A. 'a'
- B. ab
- C. true
- D. "ab"

**Câu 31.** Trong cấu trúc chương trình C++, lệnh #include dùng để làm gì?

- A. Khai báo các câu lệnh được sử dụng trong chương trình. Phải tạo các câu lệnh trước thì mới sử dụng được trong chương trình C++.
- B. Thông báo cho bộ biên dịch thêm các thư viện chuẩn trong C++. Các lệnh được sử dụng trong thân chương trình phải có prototype nằm trong các thư viện chuẩn này.
- C. Thông báo trong chương trình sử dụng các lệnh tính toán, thông báo các biến sử dụng trong thân chương trình.
- D. Không có đáp án đúng.

**Câu 32.** Trong C++ khi cần khai báo hằng ta dùng từ khóa:

- A. const

- B. var
- C. #include
- D. type

**Câu 33.** Thư viện `<iostream>` trong C++ cung cấp:

- A. Mọi tiện ích để làm việc.
- B. Các tiện ích có sẵn để làm việc với số học.
- C. Các thư viện chuẩn trong C++.
- D. Các tiện ích có sẵn để làm việc với bàn phím và màn hình.

**Câu 34.** Trong C++ khai báo hằng đúng là:

- A. `PI=3.14`
- B. `const PI=3.14`
- C. `const PI:=3.14`
- D. `const 3.14`

**Câu 35.** Chọn phát biểu đúng nhất khi nói về biến?

- A. Biến là đại lượng nhận giá trị trước khi chương trình thực hiện.
- B. Biến là đại lượng được đặt tên và có giá trị thay đổi trong quá trình thực hiện chương trình.
- C. Biến có thể lưu trữ nhiều loại giá trị khác nhau.
- D. Biến có thể đặt hoặc không đặt tên gọi.

**Câu 36.** Bài toán giải phương trình bậc nhất  $ax+b=0$  có các biến là:

- A. a,b,x
- B. a,b
- C. x
- D. b,x

**Câu 37.** Chú thích trên một dòng nào sau đây là chính xác?

- A. `\\lap trinh c++`
- B. `{lap trinh c++}`
- C. `//lap trinh c++`
- D. `/*lap trinh c++`

**Câu 38.** Cho chương trình sau:

```
#include <iostream>
int main()
{
    count<<"lap trinh c++";
}
```

Chọn phát biểu đúng khi nói về chương trình trên:

- A. Phần khai báo chỉ có một câu lệnh khai báo thư viện.
- B. Chương trình không có phần khai báo.
- C. Chương trình không có hàm main().
- D. Không có câu lệnh nào trong phần thân chương trình.

**Câu 39.** Khi khảo sát phương trình đường thẳng  $ax+by+c=0$ , các hệ số a, b, c có thể được khai báo:

- A. Là những hằng số học.
- B. Là những biến đơn.
- C. Là những hằng xâu.
- D. Là những hằng logic.

**Câu 40.** Chương trình C++ là một tệp văn bản có đuôi mặc định là:

- A. .pas
- B. .cpp
- C. .exe
- D. .doc

**Câu 41.** Khẳng định nào sau đây là đúng khi nói về hàm main() ?

- A. Hàm main() là nơi kết thúc mọi hoạt động của chương trình.
- B. Hàm main() là nơi khai báo các chương trình con khác.
- C. Hàm main() là nơi khai báo các thư viện.
- D. Hàm main() là nơi bắt đầu mọi hoạt động của chương trình.

**Câu 42.** Trong cấu trúc chương trình C++ có bao nhiêu hàm main() ?

- A. 1
- B. 2
- C. 3
- D. 4

**Câu 43.** Kết thúc một dòng lệnh trong chương trình C++, ta sử dụng ký hiệu gì?

- A. Dấu chấm (.)
- B. Dấu chấm phẩy (;)
- C. Dấu hai chấm (:)
- D. Dấu phẩy (,)

**Câu 44.** Khẳng định nào sau đây là đúng khi nói về hằng ?

- A. Khai báo hằng thường được sử dụng cho những giá trị cố định mà xuất hiện nhiều lần trong chương trình.
- B. Khai báo hằng thường được sử dụng cho những giá trị thay đổi trong chương trình.
- C. Khai báo hằng được sử dụng thay cho khai báo biến.
- D. Khai báo hằng nhất định phải có khi viết chương trình.

**Câu 45.** Sau hàm main() là dấu:

- A. }
- B. [
- C. {
- D. ]

**Câu 46.** Biến t nhận các giá trị 1 ; 2 ; 3 thì t thuộc kiểu:

- A. int
- B. char
- C. float
- D. bool

**Câu 47.** Biến t nhận các giá trị 1.3; 2.3 thì t thuộc kiểu:

- A. int
- B. char
- C. float
- D. bool

**Câu 48.** Biến s nhận các giá trị 'a','b','c' thì s thuộc kiểu:

- A. int
- B. char
- C. float
- D. bool

**Câu 49.** Phạm vi giá trị của kiểu bool là :

- A. 0-255
- B. true/false
- C. 0- $10^9$
- D. 255- $10^9$

**Câu 50.** Bộ nhớ lưu trữ một giá trị của kiểu bool là:

- A. 2 byte
- B. 3 byte
- C. 4 byte
- D. 1 byte

**Câu 51.** Số b được dùng để tính toán các số nguyên trong phạm vi từ  $-2^{15}$  đến  $2^{15}-1$  thì b thuộc kiểu:

- A. bool
- B. char
- C. short
- D. float

**Câu 52.** Số c được dùng để lưu tổng  $(2.5+1.2)$  thì c thuộc kiểu:

- A. bool
- B. float
- C. int
- D. char

**Câu 53.** Số z được dùng để chứa 2 giá trị true hoặc false thì z thuộc kiểu:

- A. bool
- B. float
- C. int
- D. char

**Câu 54.** Biến s được dùng để chứa các kí tự trong phạm vi từ 'a' đến 'z' thì biến s thuộc kiểu:

- A. bool
- B. float
- C. int
- D. char



**Câu 55.** Xét khai báo biến:

int a,b;

float c;

Chương trình dịch trong C++ sẽ cấp phát bao nhiêu byte bộ nhớ cho các biến trong khai báo trên:

A. 13 byte

B. 12 byte

C. 8 byte

D. 3 byte

**Câu 56.** Biến a có thể nhận các giá trị 3; 200; 150 và biến b có thể nhận các giá trị 0.2; 0.3; 10.99. Khai báo nào trong các khai báo sau là đúng nhất?

A. int a,b;

B. float a,b;

C. int a;  
float b;

D. float a;  
int b;

**Câu 57.** Trong ngôn ngữ lập trình C++, kiểu dữ liệu nào trong các kiểu sau có bộ nhớ lưu trữ lớn nhất?

A. int

B. char

C. short

D. bool

**Câu 58.** Phạm vi giá trị của kiểu char thuộc:

A. Từ 0 đến 255

B. Từ  $-2^{15}$  đến  $2^{15} - 1$

C. Từ 0 đến  $2^{16} - 1$

D. Từ  $-2^{31}$  đến  $2^{31} - 1$

**Câu 59.** Phạm vi giá trị của kiểu int thuộc:

A. Từ 0 đến 255

B. Từ  $-2^{15}$  đến  $2^{15} - 1$

C. Từ 0 đến  $2^{16} - 1$

D. Từ  $-2^{31}$  đến  $2^{31} - 1$

**Câu 60.** Phạm vi giá trị của kiểu short thuộc:

A. Từ 0 đến 255

B. Từ  $-2^{15}$  đến  $2^{15} - 1$

C. Từ 0 đến  $2^{16} - 1$

D. Từ  $-2^{32}$  đến  $2^{32} - 1$

**Câu 61.** Để khai báo biến x kiểu thực ta viết:

A. float x;

B. char x;

C. int x;

D. short x;

**Câu 62.** Để khai báo biến z kiểu logic ta viết:

- A. float x;
- B. bool x;
- C. int x;
- D. short x;

**Câu 63.** Cho đoạn chương trình sau:

```
#include< iostream>
using namespace std;
int a;
main()
{
    cin>>a;
    count<< a;
    return 0;
}
```

Biến a thuộc dữ liệu kiểu:

- A. int
- B. float
- C. bool
- D. str

**Câu 64.** Cho đoạn chương trình sau:

```
#include< iostream>
using namespace std;
int a;
float b;
main()
{
    cin>>a>>b;
    count<< a;
    count<< b;
    return 0;
}
```

Biến b thuộc dữ liệu kiểu:

- A. int
- B. float
- C. bool
- D. str

**Câu 65.** Khai báo nào sau đây là đúng:

- A. short x;
- B. x:short;
- C. x=short;
- D. short:x;

**Câu 66.** Giả sử trong chương trình cần các biến thực a,b,c và các biến nguyên x,y,z. Khi đó khai báo đúng là:

- A. float a,b,c,x,y,z;
- B. int a,b,cx,y,z;
- C. float a,b,c;  
int x,y,z;
- D. float x,y,z;  
int a,b,c;

**Câu 67.** Trong ngôn ngữ lập trình C++, giả sử ta thực hiện phép tính  $x = a/b$ . Thì x phải khai báo kiểu dữ liệu?

- A. int
- B. float
- C. bool
- D. char

**Câu 68.** Danh sách các biến là một hoặc nhiều tên biến, các tên biến được viết cách nhau bởi:

- A. Dấu chấm phẩy (;)
- B. Dấu chấm (.)
- C. Dấu hai chấm (:)
- D. Dấu phẩy (,)

**Câu 69.** Trong C++, cú pháp để khai báo biến là:

- A. < kiểu dữ liệu > < danh sách biến >
- B. < kiểu dữ liệu > : < danh sách biến >
- C. < danh sách biến > : < kiểu dữ liệu >
- D. < danh sách biến > < kiểu dữ liệu >

**Câu 70.** Trong C++, khai báo biến có cú pháp như sau:

< kiểu dữ liệu > < danh sách biến >;

Khẳng định nào sau đây là đúng :

- A. Mỗi biến chỉ được khai báo một lần.
- B. Mỗi biến có thể được khai báo nhiều lần.
- C. Các biến không cần phải khai báo.
- D. Mỗi biến chỉ cần gán giá trị không cần phải khai báo.

**Câu 71.** Trong C++, khai báo biến có cú pháp như sau:

< kiểu dữ liệu > < danh sách biến >;

Khẳng định nào sau đây là đúng :

- A. Danh sách biến là một hoặc nhiều tên biến, các tên biến được viết cách nhau bởi dấu chấm phẩy “;”
- B. Danh sách biến là một hoặc nhiều tên biến, các tên biến được viết cách nhau bởi dấu phẩy “,”
- C. Danh sách biến là một biến.
- D. Danh sách biến là các hằng số.

**Câu 72.** Khai báo nào sau đây đúng?

- A. float a b;
- B. float a,b;
- C. a,b:float;

**D.** a,b float;

**Câu 73.** Đại lượng dùng để lưu trữ giá trị và giá trị có thể được thay đổi trong quá trình thực hiện chương trình gọi là:

- A.** Hằng
- B.** Hàm
- C.** Biểu thức
- D.** Biến

**Câu 74.** Biến c có phạm vi giá trị là  $[-2^{64}, 2^{63}-1]$ . Cách khai báo nào sau đây là đúng :

- A.** char c ;
- B.** short c ;
- C.** long long c ;
- D.** int c ;

**Câu 75.** Khẳng định nào sau đây là **sai** ?

- A.** Cấu trúc khai báo biến có thể xuất hiện nhiều lần và ở vị trí bất kỳ trong chương trình.
- B.** Danh sách biến là một hoặc nhiều tên biến, các tên biến được viết cách nhau bởi dấu phẩy.
- C.** Trong C++ cho phép ta vừa khai báo biến vừa khởi tạo giá trị cho biến.
- D.** Biến là đại lượng không thay đổi trong quá trình thực hiện chương trình.

**Câu 76.** Biểu thức  $[(x+y)*z]-(x^2-y^2)$  chuyển sang C++ là:

- A.**  $((x+y)*z)-(x^2-y^2)$
- B.**  $((x+y)*z)-(x*x-y*y)$
- C.**  $((x+y)*z)-(x^2-y^2)$
- D.**  $(x+y)*z-x*x-y*y$

**Câu 77.** Trong phép toán số học với số nguyên phép toán lấy phần dư trong C++ là:

- A.** %
- B.** mod
- C.** /
- D.** div

**Câu 78.** Trong phép toán số học với số nguyên phép toán lấy phần nguyên trong C++ là:

- A.** %
- B.** mod
- C.** /
- D.** div

**Câu 79.** Trong phép toán quan hệ phép so sánh bằng trong C++ được viết là:

- A.** ==
- B.** =
- C.** <=
- D.** >=

**Câu 80.** Trong phép toán quan hệ phép so sánh lớn hơn hoặc bằng trong C++ được viết là:

- A. ==
- B. =
- C. <=
- D. >=

**Câu 81.** Trong phép toán quan hệ phép khác trong C++ được viết là:

- A. ==
- B. !=
- C. #
- D. <>

**Câu 82.**  $x^2$  được biểu diễn trong C++ là:

- A. x\*\*2
- B. x\*2
- C.  $x^2$
- D. x\*x

**Câu 83.** Trong C++ câu lệnh gán có dạng:

- A. < tên biến> := < biểu thức>
- B. < tên biến> = < biểu thức>
- C. < tên biến> == < biểu thức>;
- D. < tên biến> = < biểu thức>;

**Câu 84.** Trong C++ khi viết x++ có nghĩa là:

- A. Giảm x đi 2 đơn vị
- B. Tăng x lên 2 đơn vị
- C. Tăng x lên 1 đơn vị
- D. X giữ nguyên giá trị

**Câu 85.** Biểu thức a+b chuyển sang C++ là:

- A. sqrt(a+b)
- B. sqr(a+b)
- C. math.sqrt(a+b)
- D. a+b

**Câu 86.** Trong C++ khi viết x=y=3 có nghĩa là:

- A. Gán cho cả x và y giá trị 3.
- B. Thay x bằng y.
- C. So sánh x và y với 3.
- D. Tăng x và y lên 3 đơn vị.

**Câu 87.** Phép ‘và’ trong C++ kí hiệu là:

- A. ||
- B. and
- C. !
- D. &&

**Câu 88.** Để thể hiện điều kiện  $5 \leq x \leq 11$ , trong C++ được viết như sau:

- A. (5<=x)&&(x<=11)

- B.  $(5 \leq x) \text{ and } (x \leq 11)$
- C.  $(5 \leq x) \text{ || } (x \leq 11)$
- D.  $(5 \leq x) \text{ or } (x \leq 11)$

**Câu 89.** Trong C++ việc nhập dữ liệu từ bàn phím được thực hiện bằng lệnh:

- A. `cin >> biến1 >> biến2 >>...>> biếnN;`
- B. `cin << biến1 << biến2 <<...<< biếnN;`
- C. `cout >> biến1 >> biến2 >>...>> biếnN;`
- D. `cout << biến1 << biến2 <<...<< biếnN;`

**Câu 90.** Để nhập vào 3 số a,b,c từ bàn phím, ta dùng lệnh sau:

- A. `cin << a << b << c;`
- B. `cout >> a >> b >> c;`
- C. `cout << a << b << c;`
- D. `cin >> a >> b >> c;`

**Câu 91.** Cho đoạn chương trình sau:

```
#include <iostream>
using namespace std;
int main()
{
    cout<<"xin chào";
    return 0;
}
```

Trên màn hình sẽ hiển thị kết quả:

- A. Xin chào
- B. Không chạy được vì có lỗi
- C. 'Xin chào'
- D. "Xin chào"

**Câu 92.** Để đưa dữ liệu ra màn hình, C++ cung cấp đối tượng chuẩn:

- A. `cout << kq1 << kq2 << ... << kqN << endl;`
- B. `cint << kq1 << kq2 << ... << kqN << endl;`
- C. `cout >> kq1 >>kq2 >> ... >>kqN >>endl;`
- D. `cint >> kq1 >>kq2 >> ... >>kqN >>endl;`

**Câu 93.** Để đưa ra màn hình số 1 và 2 trên cùng 1 dòng trong C++ ta dùng lệnh:

- A. `cout<<1<< endln ;`  
`cout<<2 ;`
- B. `cout<<1<<2 ;`
- C. `cint<<1<<2`
- D. `cout>>1>>2 ;`

**Câu 94.** Để đưa ra màn hình số 1 và 2 trên 2 dòng khác nhau trong C++ ta dùng lệnh:

- A. `cout<<1<< endl ;`  
`cout<<2 ;`
- B. `cout<<1<<2 ;`
- C. `cint<<1<<2`
- D. `cout>>1>>2 ;`

**Câu 95.** Lệnh cout trong c++ có tác dụng gì?

- A. Là stream đầu ra chuẩn trong C++
- B. Là lệnh chú thích trong C++
- C. Là stream đầu vào chuẩn trong C++
- D. Là lệnh khai báo một biến

**Câu 96.** Lệnh cin trong c++ có tác dụng gì?

- A. Là stream đầu ra chuẩn trong C++
- B. Là lệnh chú thích trong C++
- C. Là stream đầu vào chuẩn trong C++
- D. Là lệnh khai báo một biến

**Câu 97.** Lệnh cout trong C++ đi kèm với dấu nào:

- A. //
- B. <<
- C. >>
- D. &&

**Câu 98.** Lệnh cint trong C++ đi kèm với dấu nào:

- A. //
- B. <<
- C. >>
- D. &&

**Câu 99.** Cho đoạn chương trình sau:

```
#include <iostream>
using namespace std;
int a,b;
int main()
{
    cin >>a>>b;
    cout<< a<< endl;
    cout<<b;
    return 0;
}
```

Nếu nhập a=2, b=3 thì trên màn hình sẽ hiển thị:

- A. 2 3
- B. 3 2
- C. 2  
3
- D. Không có giá trị.

**Câu 100.** Để in ra màn hình dòng chữ “lập trình C++” ta sử dụng câu lệnh:

- A. cout<<“lập trình C++” ;
- B. cout>>“lập trình C++” ;
- C. cint<<“lập trình C++” ;
- D. cout “lập trình C++” ;

**Câu 101.** Để mở 1 trang soạn thảo mới ta chọn:

- A. Menu File → New File (CTRL + N)
- B. Menu File → File Open (CTRL + O)
- C. Menu File → Save As
- D. Menu File → Save (CTRL + S)

**Câu 102.** Để lưu chương trình vào đĩa ta chọn:

- A. Menu File → New File (CTRL + N)
- B. Menu File → File Open (CTRL + O)
- C. Menu File → Close
- D. Menu File → Save (CTRL + S)

**Câu 103.** Để đóng dự án ta chọn:

- A. File → Close Project.
- B. File → Save
- C. File → Save as
- D. File → New

**Câu 104.** Sau khi soạn thảo xong chương trình, để dịch và chạy thử chương trình, ta nhấn phím:

- A. Ctrl + F9
- B. F3
- C. F9
- D. F10

**Câu 105.** Sau khi dự án chương trình đã được dịch thành công, để chạy thử chương trình ta chọn :

- A. ALT + F2
- B. Save
- C. Ctrl + F3
- D. F9

**Câu 106.** Để thoát khỏi môi trường Code::Blocks ta nhấn tổ hợp phím:

- A. ALT + F4
- B. File → Save
- C. Ctrl + N
- D. Ctrl + F10

**Câu 107.** Phím tắt F9 trong CodeBlocks dùng để thay cho lệnh:

- A. Build → Run
- B. Build → Build and run
- C. File → Exit
- D. File → Close Project

**Câu 108.** Để mở file đã có trên đĩa ta dùng tổ hợp phím:

- A. ALT+F4
- B. Ctrl+O
- C. Ctrl+S
- D. Ctrl+N



**Câu 109.** Để lưu lại file trên đĩa ta chọn tổ hợp phím:

- A. ALT+F4
- B. Ctrl+O
- C. Ctrl+S
- D. Ctrl+N

**Câu 110.** Để lưu lại file trên đĩa với tên gọi khác ta chọn:

- A. File → Save As
- B. File → Save
- C. Ctrl+S
- D. File → New

**Câu 111.** Phím tắt CTRL + F10 trong Code::Blocks dùng để thay cho lệnh:

- A. Build → Run
- B. Build → Build and run
- C. File → Exit
- D. File → Close Project

**Câu 112.** Cách nào sau đây không thoát khỏi môi trường Code::Blocks?

- A. ALT + F4
- B. Build → Build and run
- C. File → Quit
- D. CTRL + Q

**Câu 113.** Tổ hợp phím Ctrl+Q trong CodeBlocks dùng để thay cho lệnh:

- A. Build → Run
- B. Build → Build and run
- C. File → Exit
- D. File → Close Project

**Câu 114.** Tổ hợp phím ALT + F4 có chức năng tương tự tổ hợp phím nào sau đây:

- A. Ctrl + F9
- B. Ctrl + F10
- C. Ctrl + Q
- D. Ctrl + S

**Câu 115.** Cửa sổ build log bên dưới màn hình dùng để:

- A. Hiển thị và thông báo về lỗi.
- B. Thông báo về các biến.
- C. Thông báo về vị trí điểm breakpoint.
- D. Thông báo về những tùy chọn có sẵn.

**Câu 116.** Cấu trúc của rẽ nhánh dạng thiếu trong ngôn ngữ lập trình C++ là:

- A. if< điều kiện>:  
    < câu lệnh>
- B. if< điều kiện>  
    < câu lệnh>
- C. if< điều kiện> then:

< câu lệnh>

D. if (< điều kiện>) < câu lệnh>;

**Câu 117.** Cấu trúc của rẽ nhánh dạng đủ trong ngôn ngữ lập trình C++ là:

A. if< điều kiện>:

< câu lệnh 1>

else

< câu lệnh 2>

B. if< điều kiện>:

< câu lệnh 1>

else:

< câu lệnh 2>

C. if< điều kiện>

< câu lệnh 1>

else:

< câu lệnh 2>

D. if(< điều kiện>)

< câu lệnh 1>;

else

< câu lệnh 2>;

**Câu 118.** Trong cấu trúc rẽ nhánh dạng thiếu câu lệnh < câu lệnh> được thực hiện khi:

A. Điều kiện sai.

B. Điều kiện đúng.

C. Điều kiện bằng 0.

D. Điều kiện khác 0.

**Câu 119.** Trong cấu trúc rẽ nhánh dạng đủ câu lệnh < câu lệnh 1> được thực hiện khi :

A. Điều kiện sai.

B. Điều kiện đúng.

C. Điều kiện bằng 0.

D. Điều kiện khác 0.

**Câu 120.** Trong cấu trúc rẽ nhánh dạng đủ câu lệnh < câu lệnh 2> được thực hiện khi :

A. Điều kiện sai.

B. Điều kiện đúng.

C. Điều kiện bằng 0.

D. Điều kiện khác 0.

**Câu 121.** Trong các phát biểu sau, phát biểu nào sử dụng câu lệnh rẽ nhánh dạng thiếu:

A. a là số chẵn.

B. Điều kiện cần để a là số chẵn là a chia hết cho 2.

C. a là số chẵn khi a chia hết cho 2.

D. Nếu a chia hết cho 2 thì a là số chẵn.

**Câu 122.** Trong các phát biểu sau, phát biểu nào sử dụng câu lệnh rẽ nhánh dạng đủ:

A. Nếu a chia hết cho 2 thì a là số chẵn, ngược lại a là số lẻ.

B. Số a chia hết cho 2 thì a là số chẵn.

C. a là số chẵn khi a chia hết cho 2.

**D.** Điều kiện cần để a là số chẵn là a chia hết cho 2.

**Câu 123.** Cho đoạn chương trình sau:

```
a=2
b=3
if (a>b)
    a=a*2;
else
    b=b*2;
```

Sau khi thực hiện đoạn chương trình trên giá trị của b là:

- A.** 4
- B.** 2
- C.** 6
- D.** Không xác định

**Câu 124.** Để đưa ra số lớn nhất trong 2 số a, b ta viết như sau:

- A.** if a < b  
    cout<< a;  
else  
    cout<< b;
- B.** if a = b  
    cout<< a;  
else  
    cout<< b;
- C.** if a > b  
    cout<< a;  
else  
    cout<< b;
- D.** if a > b  
    cout<< a;

**Câu 125.** Cho đoạn chương trình sau:

```
x=10
y=3
d=0
if x%y==0
    d=x/y
```

Sau khi thực hiện đoạn chương trình trên giá trị của d là:

- A.** 3
- B.** 1
- C.** 0
- D.** Không xác định

**Câu 126.** Câu lệnh ghép được viết trong cặp ngoặc:

- A.** []
- B.** {}
- C.** ()
- D.** <>

**Câu 127.** Câu lệnh rẽ nhánh dạng thiếu và dạng đủ điều kiện là:

- A. Nhiều câu lệnh.
- B. Biểu thức số học.
- C. Biểu thức quan hệ hoặc logic.
- D. Một câu lệnh.

**Câu 128.** Trong ngôn ngữ lập trình C++, phát biểu nào sau đây là đúng với câu lệnh rẽ nhánh dạng đủ:

- A. Nếu sau else muốn thực hiện nhiều câu lệnh thì các câu lệnh phải đặt giữa hai dấu ngoặc đơn ().
- B. Nếu sau else muốn thực hiện nhiều câu lệnh thì các câu lệnh phải đặt giữa Begin và End.
- C. Nếu sau else muốn thực hiện nhiều câu lệnh thì các câu lệnh phải đặt giữa hai cặp dấu ngoặc nhọn ({}).
- D. Nếu sau else muốn thực hiện nhiều câu lệnh thì các câu lệnh phải đặt giữa hai dấu ngoặc vuông ([]).

**Câu 129.** Trong ngôn ngữ lập trình C++, muốn kiểm tra đồng thời cả ba giá trị của a, b, c có cùng lớn hơn 0 hay không, câu lệnh if viết đúng là:

- A. if (a>0)and (b>0) and(c>0)
- B. if (a>0)or (b>0) or(c>0)
- C. if (a>0) && (b>0) &&(c>0)
- D. if (a,b,c>0)

**Câu 130.** Cấu trúc lặp với số lần biết trước trong C++ có dạng:

- A. for ([khởi tạo]; [điều kiện lặp]; [bước nhảy]) [lệnh];
- B. for ([khởi tạo]; [điều kiện lặp])[lệnh];
- C. for ([khởi tạo]; [điều kiện lặp]; [bước nhảy])
- D. for ([điều kiện lặp]; [bước nhảy]) [lệnh];

**Câu 131.** Cấu trúc lặp với số lần chưa biết trước trong C++ có dạng:

- A. while (< điều kiện lặp>) < câu lệnh>;
- B. while < câu lệnh>;
- C. while (< điều kiện lặp>) < câu lệnh>;
- D. while (< điều kiện lặp>)

**Câu 132.** Cho đoạn lệnh sau:

```
for (int i = 1; i <= 5; i++) cout<< i;
```

Trên màn hình có các giá trị là:

- A. 0 1 2 3 4 5
- B. 1 2 3 4 5
- C. 0 1 2 3 4
- D. 1 2 3 4

**Câu 133.** Cho đoạn lệnh sau:

```
for (int i = 5; i >=0; i--) cout<< i;
```

Trên màn hình có các giá trị là:

- A. 0 1 2 3 4 5
- B. 5 4 3 2 1

- C. 1 2 3 4 5
- D. 5 4 3 2 1 0

**Câu 134.** Cho đoạn lệnh sau:

```
i=1;
n=5;
while (i<=n)
    {
        cout<< i;
        i=i+1;
    }
```

Trên màn hình có các giá trị là:

- A. 0 1 2 3 4 5
- B. 5 4 3 2 1
- C. 1 2 3 4 5
- D. 5 4 3 2 1 0

**Câu 135.** Cho đoạn chương trình sau:

```
cin >> a;
n = 0;
double s = 1.0/a; // Bước 1
while(!(1.0/(a+n) < 0.0001))
    {
        n++;
        s += 1.0/(a+n);
    }
```

Trong đoạn chương trình trên vòng lặp thực hiện lặp bao nhiêu lần?

- A. a lần
- B. 10 lần
- C. n lần
- D. Không biết trước

**Câu 136.** Cho đoạn chương trình sau:

```
s=0
for ( i =1;i<=5;i++) s=s+i;
```

Sau khi thực hiện đoạn chương trình trên giá trị của s là:

- A. 1
- B. 15
- C. 6
- D. 21

**Câu 137.** Khẳng định nào sau đây là đúng:

- A. Mọi bài toán giải được với vòng lặp for... đều có thể giải được với vòng lặp while.
- B. Mọi bài toán giải được với vòng lặp for... đều không thể giải được với vòng lặp while.
- C. Mọi bài toán giải được với vòng lặp while đều có thể giải được với vòng lặp for.
- D. Một bài toán chỉ có thể giải được với một vòng lặp.

**Câu 138.** Cho đoạn chương trình sau:

for (i=1;i<=10;i++) cout<< i;  
Vòng lặp trên sẽ lặp bao nhiêu lần:

- A. 9 lần
- B. 10 lần
- C. Không lặp
- D. 11 lần

**Câu 139.** Cho đoạn chương trình sau:

```
s=0  
i=1  
while (i<=5)  
{  
    s=s+1;  
    i=i+1;  
}
```

Sau khi thực hiện đoạn chương trình trên giá trị của s là:

- A. 9
- B. 15
- C. 5
- D. 10

**Câu 140.** Trong cấu trúc: *for ([khởi tạo]; [điều kiện lặp]; [bước nhảy]) [lệnh];*

Vòng lặp kết thúc khi nào?

- A. Sau khi điều kiện lặp trở thành false.
- B. Sau khi điều kiện lặp trở thành true.
- C. Sau khi điều kiện lặp không xác định.
- D. Sau khi điều kiện lặp bằng 0.

**Câu 141.** Khẳng định nào sau đây là đúng trong cấu trúc:

*for ([khởi tạo]; [điều kiện lặp]; [bước nhảy]) [lệnh];*

- A. Bắt buộc phải viết phần khởi tạo, dấu chấm phẩy không nhất thiết phải có.
- B. Không bắt buộc phải viết phần khởi tạo, nhưng dấu chấm phẩy phải có.
- C. Bắt buộc phải viết phần khởi tạo, dấu chấm phẩy phải có.
- D. Không bắt buộc phải viết phần khởi tạo và dấu chấm phẩy không cần phải có.

**Câu 142.** Trong cấu trúc *while (< điều kiện lặp>) < câu lệnh>;* Điều kiện là:

- A. Nhiều câu lệnh
- B. Biểu thức số học.
- C. Biểu thức quan hệ hoặc logic.
- D. Một câu lệnh.

**Câu 143.** Khẳng định nào sau đây là đúng:

- A. Việc lặp với số lần chưa biết trước sẽ chỉ kết thúc khi một điều kiện cho trước được thoả mãn.
- B. Việc lặp với số lần chưa biết trước sẽ chỉ kết thúc khi một điều kiện cho trước không được thoả mãn.
- C. Việc lặp với số lần chưa biết trước sẽ chỉ kết thúc khi một điều kiện cho trước bằng 0.
- D. Việc lặp với số lần chưa biết trước sẽ chỉ kết thúc khi một điều kiện cho trước không xác định.

**Câu 144.** Trong vòng lặp `for(i=10;i>=0;i--)`

Giá trị của `i`:

**A.** Tăng dần

**B.** Giảm dần

**C.** Không xác định

**D.** Giữ nguyên