**Câu hỏi ôn tập môn lập trình C**

**Câu 1 :**

* Khái niệm:
* stdin (standard input): Dòng vào tiêu chuẩn, lấy thông tin từ các thiết bị nhập. Các hàm scanf, gets, getchar,… đều nhận dữ liệu từ stdin.
* stdout (standard output): Dòng ra tiêu chuẩn. Các hàm printf, puts, putchar,… lấy dữ liệu từ dòng ra stdout để đưa ra thiết bị đầu ra.
* stdprn (standard print): Dòng ra máy in tiêu chuẩn. Máy in nhận dữ liệu từ stdprn.

**Câu 2:**

* Câu lệnh if là câu lệnh kiểm tra điều kiện, nếu điều kiện được đưa ra đúng, khối lệnh sau if sẽ được thực hiện, nếu không, sẽ chuyển sang câu lệnh tiếp theo.  
  Câu lệnh switch là câu lệnh kiểm tra điều kiện theo giá trị của 1 biến. Nếu giá trị của biến trùng với lệnh case nào thì các lệnh sau case sẽ được thực hiện.

Trong 1 chương trình:

* If có thể thay thế switch.
* Switch không thể thay thế cho if (vì switch-case chỉ kiểm tra với điều kiện là 1 giá trị cụ thể)

**Câu 3:**

* Cú pháp câu lệnh for:

For (khởi tạo giá trị biến đếm ; điều kiện thực hiện ; tăng giảm biến đếm) {

// Khối lệnh

}

Thực hiện:

* Biến đếm được khởi tạo với giá trị cho trước
* Nếu điều kiện đúng, khối lệnh được thực thi
* Sau khi thực thi khối lệnh, biến đếm được tăng/giảm
* Kiểm tra điều kiện, nếu đúng, vòng lặp tiếp tục, nếu sai, dừng vòng lặp và thực hiện các lệnh tiếp theo

**Câu 4:**

* Cú pháp lệnh while:

While (điều kiện) {

// Khối lệnh

}

Thực hiện:

* Kiểm tra điều kiện, nếu đúng, khối lệnh được thực hiện, nếu sai, bỏ qua vòng lặp

**Câu 5:**

* Cú pháp lệnh do-while:

Do {

// Khối lệnh

} while (điều kiện);

Thực hiện:

* Thực hiện khối lệnh trong do
* Kiểm tra điều kiện, nếu đúng, tiếp tục thực hiện khối lệnh trong do, nếu sai, thoát khỏi vòng lặp.

**Câu 6:**

* Lệnh continue: Bỏ qua lần lặp hiện tại, chuyển sang vòng lặp kế tiếp

Lệnh break: Thoát khỏi vòng lặp

Lệnh return: Trả về giá trị sau khi kết thúc vòng lặp hoặc hàm

**Câu 7:**

* Hàm định nghĩa sẵn: Phải khai báo thư viện mở rộng để sử dụng được hàm

Hàm tự định nghĩa: Phải khai báo tên hàm, viết các câu lệnh hàm sẽ thực hiện

Đặc điểm:

* Chia nhỏ phần thân chương trình
* Dễ đọc
* Có thể sử dụng lại nhiều lần

**Câu8:**

|  |  |  |
| --- | --- | --- |
| Đặc điểm | Biến cục bộ | Biến toàn cục |
|  | Khai báo bên trong 1 hàm | Khai báo bên ngoài các hàm |
| Thời gian tồn tại | Khi thực thi hàm đó | Trong toàn bộ chương trình |
| Phạm vi sử dụng | Chỉ dùng trong hàm mà biến được khai báo | Dùng bất kì đâu trong chương trình |
| Thời điểm cấp phát bộ nhớ | Khi hàm được gọi và chạy | Khi chương trình bắt đầu |

* Phân biệt:

Biến được khai báo trong main() là biến cục bộ.

* Khi thực hiện các lệnh trong 1 hàm, biến cục bộ có cùng tên với biến toàn cục thì giá trị sẽ được lấy theo biến cục bộ.
* Biến tĩnh là biến được định kiểu và kích thước. Biến tĩnh được cấp phát 1 phần bộ nhớ xác định mà không cần biết sẽ dùng hết dung lượng đó hay không. Biến tĩnh và vùng nhớ của nó sẽ tồn tại đến khi chương trình kết thúc.

Biến động là 1 biến con trỏ, không được cấp phát sẵn bộ nhớ. Khi thực hiện, biến động sẽ được cấp phát lượng bộ nhớ tùy theo người lập trình. Vùng nhớ và địa chỉ ô nhớ của biến động có thể thay đổi.

**Câu 11:**

* Đệ quy thường chạy lâu hơn vòng lặp do nó tốn dung lượng bộ nhớ sau mỗi lần chạy hàm.

**12**

* Tham số biến: Hàm chỉ nhận vào giá trị của biến, và hàm không thể thay đổi giá trị của biến

Tham số con trỏ: Hàm nhận vào địa chỉ của ô nhớ chứa dữ liệu, hàm có thể làm thay đổi giá trị

**13:**Dung lượng bộ nhớ cho con trỏ \*p1 và \*p2 là như nhau vì cả 2 con trỏ đều chưa được cấp phát dữ liệu.

**14:**Truyền con trỏ là chỉ truyền giá trị của con trỏ, mọi thay đổi chỉ tác động lên giá trị này chứ không làm thay đổi giá trị trong ô nhớ mà con trỏ trỏ tới.

Truyền địa chỉ con trỏ có thể làm thay đổi giá trị được lưu trong ô nhớ mà con trỏ trỏ tới.

**15:**Khi cấp phát bộ nhớ, con trỏ sẽ nhận địa chỉ của ô nhớ một cách ngẫu nhiên. Việc cấp phát sẽ dọn sạch dữ liệu trên các ô nhớ được cấp phát.

Int \*p;

P = (int \*)malloc(100\*sizeof(int));

**16:**Chuyển kiểu ngầm định: Trình biên dịch sẽ tự động chuyển kiểu nếu kiểu dữ liệu truyền cho biến không phù hợp với kiểu dữ liệu của biến đó. Trong các tính toán, để đảm bảo các tham số phải cùng kiểu, trình biên dịch sẽ chuyển kiểu dữ liệu chiếm ít bộ nhớ lên kiểu dữ liệu chiếm nhiều bộ nhớ: int à unsigned int à long int à unsigned long int à float à double à long double

Chuyển kiểu tường minh: Chuyển từ 1 kiểu dữ liệu này sang kiểu dữ liệu khác theo ý muốn của người lập trình: (kiểu dữ liệu) dữ liệu cần chuyển

**17:**Khi tăng giảm địa chỉ trên mảng 2 chiều thì sẽ chỉ đến phần tử ở hàng/cột trước hoặc sau nó.

**18:**Chương trình con có tham số là mảng 1 chiều thì truyền vào tên mảng.

Chương trình con trỏ về 1 mảng 1 chiều chính là hàm trả về 1 con trỏ.

<kiểu dữ liệu> \*<tên hàm>(<các tham số>) {

// Các câu lệnh

return (\*<con trỏ>);

}

**19:**Kích thước: Mảng Num1: 20 byte; Mảng Num2: 10 byte

Không thể gán Num1 = Num2 vì 2 mảng không cùng kích thước.

* ..
* <bỏ>

**22:**Gán p = Num là sai vì không thể gán 1 mảng chi địa chỉ 1 con trỏ, chúng không có cùng kích thước dữ liệu

Gán p = (int \*)a là đúng. Con trỏ p sẽ có giá trị bằng giá trị của biến a (nếu a khác kiểu p thì việc ép kiểu sẽ xảy ra)

**23:**Khai báo con trỏ hàm:

<kiểu dữ liệu> \*<tên hàm>(<tham số>) {

// Các câu lệnh

}

Ta có thể gọi giá trị mà hàm trỏ tới trên ô nhớ.

**24:**Union thường được khai báo bên trong struct và nó chỉ toond bộ nhớ khi được dùng tới. Union được dùng khi khai báo các struct mà các trường trong nó thay đổi tùy thuộc vào trạng thái.

**25:**Khai báo struct tự trỏ:

Struct <tên struct> {

// Các trường dữ liệu

Struct <tên struct> \*tiep;

**};**

**26;**Kiểu nhị phân lưu dữ liệu ra dưới dạng 1 tệp nhị phân, kết thúc tệp bằng EOF (có giá trị bằng -1)

Kiểu văn bản chỉ khác kiểu nhị phân khi xử lí kí tự xuống dòng. Đối với các kí tự khác thì đọc ghi như nhau.

**27:**Các kiểu mở tệp:

|  |  |
| --- | --- |
| Mode | Mô tả |
| r , rb | Mở một file có sẵn để đọc |
| w , wb | Tạo file mới để ghi, nếu file có sẵn thì sẽ bị ghi mới hoàn toàn |
| a, ab | Mở một file để ghi từ vị trí cuối cùng của file, nếu file không tồn tại sẽ tạo mới |
| r+, rb+ | Mở một file có sẵn để đọc và ghi |
| w+, wb+ | Tạo file mới để đọc và ghi, nếu file có sẵn thì sẽ bị ghi mới hoàn toàn |
| a+, ab+ | Mở một file để để đọc và ghi. Có thể đọc từ đầu file, nhưng khi ghi thì ghi từ vị trí cuối cùng của file, nếu file không tồn tại sẽ tạo mới |

Các chế độ để xử lý file nhị phân: “rb”, “wb”, “ab”, “rb+”, “wb+”, “ab+”, “r+b”, “a+b”

**28:**Khi mở tệp có bổ sung thì con trỏ tệp sẽ trỏ vào cuối tệp. Sau khi đọc, ghi, con trỏ sẽ ở vị trí EOF.

Anh công biết ui lại còn phải dặn à

Phương anh à anh nhớ em nhiều lắm em ạ

Dạo này em đang làm gì nhớ anh không anh không biết có nên im lặng như thế này lâu như thế này không nữa không

Anh nhớ em nhiều nhiều lắm em có biết không anh thực sự muốn gặp em ngay bây h em có biết không hả phương anh.Anh muốn chạy thật nhanh đến bên em và nói Anh yêu em……..rất nhiều nhiều nhiều luôn em có biết không nhưng anh không nói cho em biết đến nên anh tin là em không biết anh yêu em có đúng không.Phương anh à chưa bao giờ anh laj có một ngươi làm anh say mê đến thế lại yêu và nhớ đến thế về người đó vậy không biết nữa,thật vậy từ trước tới giờ anh cũng thích nhiều người ui nhưng chi có em thui là làm cho anh phai thao thức đêm hôm nhớ mong từng ngày em có biết không anh nhớ em quá,anh yêu em nhiều.

Trong những người mà tôi yêu quy tôi thích nhât là phương anh

Phương anh đã làm cho tôi có một cảm giác về tình yêu thực sự,mọi người biết không tui thực sự giờ đây vừa đang lo thi học lập trinh C với lập trinh hợp ngữ

Nhiều điều nữa về học tập tui học rất nhiều có phải tôi không học đâu ngày nào cũng học full HD mà sao tui vẫn học rốt nhất lớp hả mọi người.Hãy cho tui một lời nói.hay tui có nên thay đổi cách học không các bạn, có phải cách học của tôi không hiệu quả thật sự luôn.Tui có nên thay đổi cách học tập không.Mà học thì học kiểu gì để cho việc học của tôi có hiệu quả nhất trong những ngày tới đây hả các bạn của tôi ơi.Tôi Xin các bạn hãy cho tôi một lời khuyên có được không hả mọi người các bạn cua tôi đi đâu hết ui

Phương anh à em hãy cho anh một lời khuyên học như thế nào cho hiệu quả đây.Hãy đến bên anh sưởi ấm lòng anh đi em hãy đến bên anh em nhé anh nhớ em nhiều .Từ buổi cưới hôm đó mình chưa gặp nhau đâu em nhỉ

Có đúng không em,em có biết chúng ta không nói chuyện với nhau bao lậu ui không em anh nhớ em quá muốn nói chuyện với em thật nhiều muốn gần bên em thật nhiều phương anh ơi em có nhớ tới anh không anh chỉ muốn hét to là Anh Yêu Em.Phương Anh ơi anh yêu em,anh yêu em,anh yêu em ,anh yêu em…………………………….

Cuộc sống của chúng ta có được ở bên nhau không em chỉ nhờ vào sự phụ thuộc của em đó em có biết không.Anh chỉ muốn nghe thật nhiều được câu nói em nói ra là em nhớ anh,em yêu anh thì chúng t đi hết con đường cùng nhau nha em,dù có khó khăn đến đâu thì cũng nên cố gắng vượt qua nha em.Anh yêu em nhiều lắm