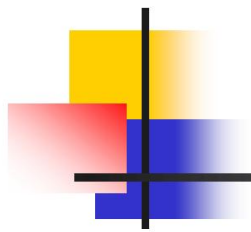


# Vòng lặp

Chỉ dành cho Trung tâm Aptech sử dụng

## Phiên 6



# Mục tiêu

---

Hiểu vòng lặp 'for' trong 'C'

Làm việc với toán tử dấu phẩy

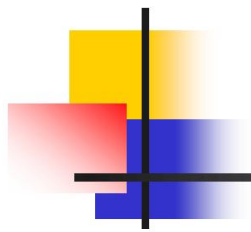
Hiểu các vòng lặp lồng nhau

Hiểu vòng lặp 'while' và 'do-while'

vòng lặp

Làm việc với các câu lệnh break và continue

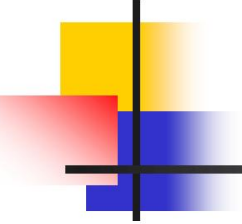
Hiểu hàm exit()



# Vòng lặp là gì?

---

Chức năng của Vòng lặp trong chương trình được sử dụng để thực thi nhiều lần cho đến khi một điều kiện cụ thể được đáp ứng



# 3 loại cấu trúc vòng lặp

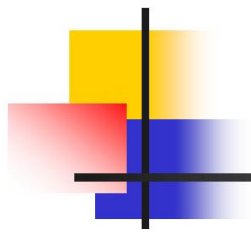
---

Vòng lặp for

Chỉ dành cho Trung tâm Aptech sử dụng

Vòng lặp while

Vòng lặp do . .while



# Vòng lặp **for** -1

---

## Cú pháp

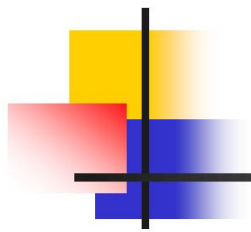
```
for (initialize counter; conditional test; re-evaluation parameter)
{
    statement
}
```

Chỉ dành cho Trung tâm Aptech sử dụng

Bộ đếm khởi tạo là một câu lệnh gán đặt biến điều khiển vòng lặp trước khi vào vòng lặp

Kiểm tra điều kiện là một biểu thức quan hệ, trong đó xác định khi nào vòng lặp sẽ thoát

Tham số đánh giá xác định cách biến điều khiển vòng lặp thay đổi mỗi khi vòng lặp được thực thi



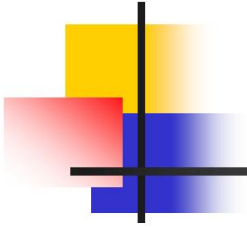
# Vòng lặp `for` -2

---

Ba phần của vòng lặp `for` phải được phân tách bằng dấu  
dấu chấm phẩy(;) )

Chỉ dành cho Trung tâm Aptech sử dụng  
Câu lệnh tạo thành phần thân của vòng lặp có thể là một câu lệnh  
đơn hoặc một câu lệnh ghép

Vòng lặp `for` tiếp tục thực thi miễn là  
kiểm tra điều kiện đánh giá là đúng. Khi điều kiện trở thành sai,  
chương trình tiếp tục thực hiện câu lệnh theo sau vòng lặp `for`



# Vòng lặp **for** -3

```
/*Chương trình này trình bày vòng lặp for trong chương trình C */  
#include <stdio.h>
```

chủ yếu()

{

int count;

printf("\tĐây là \n");

đôi với (số đếm = 1; số đếm <=6; số đếm++)

printf("\n\t\ttốt");

printf("\n\t\tthế giới. \n");

}

Chỉ dành cho Trung tâm Aptech sử dụng Ví dụ



# Toán tử dấu phẩy

Phạm vi của vòng lặp for có thể được mở rộng bằng cách bao gồm nhiều hơn một biểu thức khởi tạo hoặc tăng trong đặc tả vòng lặp for

Định dạng là : `exprn1, exprn2 ;`

Chỉ dành cho Trung tâm Aptech sử dụng

```
#include <stdio.h>
```

```
chủ yếu()
```

```
{
```

```
    int i, j ,        tối đa;
```

```
    printf("Vui lòng nhập giá trị lớn nhất \n");
```

```
    printf("bảng nào có thể được in ra: ");
```

```
    quét("%d", &max);
```

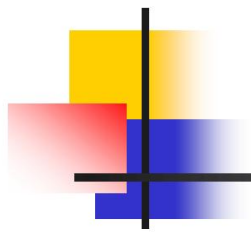
```
    đối với (i = 0, j = tối đa; i <=tối đa; i++, j--)
```

```
        printf("\n%d + %d = %d",i, j, i + j);
```

```
}
```

Ví dụ





# Vòng lặp lồng nhau-1

Vòng lặp for sẽ được gọi là vòng lặp for lồng nhau khi nó là được viết như sau

```
đối với (i = 1; i < max1; i++)  
{  
    .  
    .  
    đối với (j = 0; j < = max2; j++)  
    {  
        .  
        .  
    }  
    .  
    .  
}
```



# Vòng lặp lồng nhau-2

---

```
#include <stdio.h>
```

```
chính()
```

```
{
```

```
    int i, j, k; i =
```

```
    0;
```

Chỉ dành cho Trung tâm Aptech sử dụng

```
    printf("Nhập số hàng :"); scanf("%d", &i);
```

```
    printf("\n"); for (j = 0;
```

```
    j < i ; j++) {
```

```
        printf("\n");
```

```
        for (k = 0; k <= j; k++) /*vòng lặp for bên trong*/
```

```
        printf("*");
```

```
    }
```

```
}
```

Ví dụ



# Vòng lặp while - 1

---

Cú pháp

trong khi (điều kiện là đúng)  
Chỉ dành cho Trung tâm Aptech sử dụng  
tuyên bố ;

Vòng lặp while lặp lại các câu lệnh trong khi một điều kiện cụ thể nào đó là Đúng



# Vòng lặp while -2

---

```
/* Một chương trình đơn giản sử dụng vòng lặp while */
```

```
#include <stdio.h> main()
```

```
{ int
```

```
    đếm = 1; while( đếm
```

```
    <= 10) {
```

```
        printf("\n Đây là lần lặp %d\n",count); count++;
```

```
    }
```

```
    printf("\n Vòng lặp đã hoàn tất. \n");
```

```
    }
```

Ví dụ

Chỉ dành cho Trung tâm Aptech sử dụng



# vòng lặp `do while` -1

---

## Cú pháp

```
do{ statement; }  
while (điều kiện);
```

Chỉ dành cho Trung tâm Aptech sử dụng

Trong vòng lặp `do while` , phần thân của mã được thực thi một lần trước khi thực hiện thử nghiệm

Khi điều kiện trở thành Sai trong lệnh `do while` thì vòng lặp sẽ kết thúc và quyền điều khiển sẽ chuyển đến câu lệnh xuất hiện ngay sau câu lệnh `while`



# do.while Vòng lặp-2

---

```
#include <stdio.h> main ()  
    { int
```

```
        num1, num2; num2 = 0;
```

```
    LÂM
```

```
    {
```

```
        printf( "\nNhập một số : "); scanf("%d",&num1);  
        printf( "
```

```
            Không. là %d",num1); num2+
```

```
        +; } while
```

```
    (num1 != 0); printf ("\nTổng số
```

```
    đã nhập là %d",--num2);
```

```
/*num2 được giảm trước khi in vì số lượng cho số nguyên cuối cùng (0) không được xem xét */
```

```
    }
```

Ví dụ

Chỉ dành cho Trung tâm Aptech sử dụng



# Câu lệnh nhảy-1

---

## return sự biểu lộ

Chỉ dành cho Trung tâm Aptech sử dụng

Câu lệnh return được sử dụng để trả về từ một hàm

Nó khiến cho việc thực thi trở về điểm mà  
đã thực hiện lệnh gọi hàm

Câu lệnh return có thể có giá trị mà nó trả về cho chương trình



# Câu lệnh nhảy-2

---

## goto nhãn

Chỉ dành cho Trung tâm Aptech sử dụng  
Câu lệnh goto chuyển giao quyền điều khiển cho bất kỳ lệnh nào khác  
câu lệnh trong cùng một hàm trong chương trình C

Nó thực sự vi phạm các quy tắc của một cấu trúc chặt chẽ  
ngôn ngữ lập trình

Chúng làm giảm độ tin cậy của chương trình và khiến chương  
trình khó bảo trì





# Câu lệnh nhảy-3

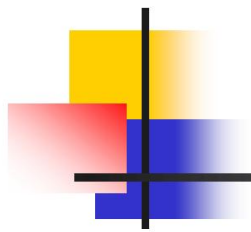
---

break tuyên bố

Câu lệnh break được sử dụng để kết thúc một trường hợp trong câu lệnh chuyển đổi

Nó cũng có thể được sử dụng để chấm dứt đột ngột một vòng lặp

Khi gặp câu lệnh break trong vòng lặp, vòng lặp được chấm dứt ngay lập tức và quyền điều khiển được chuyển đến câu lệnh theo sau vòng lặp



# câu lệnh ngắt

---

## Ví dụ

```
#include <stdio.h> chính ()
```

```
{
```

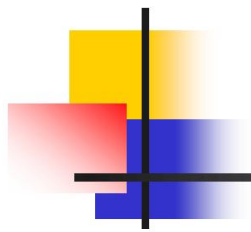
Chỉ dành cho Trung tâm Aptech sử dụng

```
int count1, count2; cho(count1
```

```
= 1, count2 = 0; count1 <= 100; count1++) {
```

```
    printf("Nhập %d count2 : ", count1); scanf("%d",  
    &count2); nếu(j==100) ngắt;
```

```
}
```



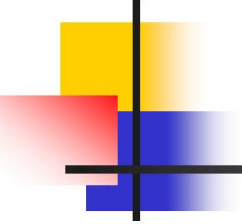
# Câu lệnh nhảy-4

---

`continue` tuyên bố

Chỉ dành cho Trung tâm Aptech sử dụng  
Câu lệnh `continue` khiến vòng lặp tiếp theo của vòng lặp bao quanh  
bắt đầu

Khi gặp câu lệnh này, phần còn lại  
các câu lệnh trong thân vòng lặp bị bỏ qua và quyền điều khiển  
được chuyển sang bước khởi tạo lại



# tiếp tục tuyên bố

---

```
#include <stdio.h> chính () {
```

Ví dụ

Chỉ dành cho Trung tâm Aptech sử dụng

```
    int số;  
    for(num = 1; num <=100; num++) {  
  
        nếu(số % 9 == 0) tiếp tục;  
        printf("%d\t",số);  
  
    }  
}
```



# Câu lệnh nhảy-5

---

`exit()` chức năng

Chỉ dành cho Trung tâm Aptech sử dụng  
`exit()` được sử dụng để thoát khỏi chương trình

Việc sử dụng chức năng này sẽ khiến chương trình bị dừng ngay lập tức và quyền kiểm soát nằm trong tay hệ điều hành