

# Tư Duy Lập Trình

<https://cybersoft.edu.vn>

# VÒNG LẶP – MẢNG - NỘI DUNG HỌC

Tại sao sử dụng vòng lặp ?

01

Vòng lặp While

02

Vòng lặp do while

03

Vòng lặp for

04

Bài tập vòng lặp

05

Mảng

06

Bài tập mảng

07

# Tại sao sử dụng vòng lặp ?

1. Viết chương trình, Nếu điểm thi của bạn <5, bạn cần phải thi lại ? Có bao nhiêu lần thi lại?
2. Nhập vào số n lớn hơn 10 để tính, nhập sai cho nhập lại ?
3. Xây dựng menu cho phép chọn , nếu nhập 1 : thực hiện chức năng x, nếu nhập 2 : thực hiện chức năng y, nhập 3 : thực hiện chức năng z, nhập 4 để thoát.
4. Tính tổng các số từ 1 - > n với n do người dùng nhập vào.



```
output  
> Điểm lần này của bạn bao nhiêu ? 4  
> Điểm lần này của bạn bao nhiêu ? 2  
> Điểm lần này của bạn bao nhiêu ? 3  
> Điểm lần này của bạn bao nhiêu ? 7  
> CHÚC MỪNG BẠN ĐÃ ĐẬU
```

# Vòng lặp



# Các loại vòng lặp

---

Vòng lặp while

Vòng lặp do....while

Vòng lặp for

Vòng lặp while

Không biết chính xác số lần lặp

## Cú pháp vòng lặp while

while (điều kiện)

{

// các dòng lệnh xử lý

}

## Cách hoạt động của vòng lặp while

Bước lặp

1

1

khởi tạo giá trị cho biến bước nhảy

2

while (điều kiện)



{

3

// các dòng lệnh xử lý

4

// tăng hoặc giảm biến bước nhảy

}

## Cách hoạt động của vòng lặp while

Bước lặp

2

Giá trị khởi gán hoặc tạo biến bước nhảy

- 
- 1 while (điều kiện)
  - 2 // các dòng lệnh xử lý
  - 3 // tăng hoặc giảm biến bước nhảy
- {  
}

## Cách hoạt động của vòng lặp while

Bước lặp

3

khởi tạo giá trị cho biến bước nhảy



1

while (điều kiện)



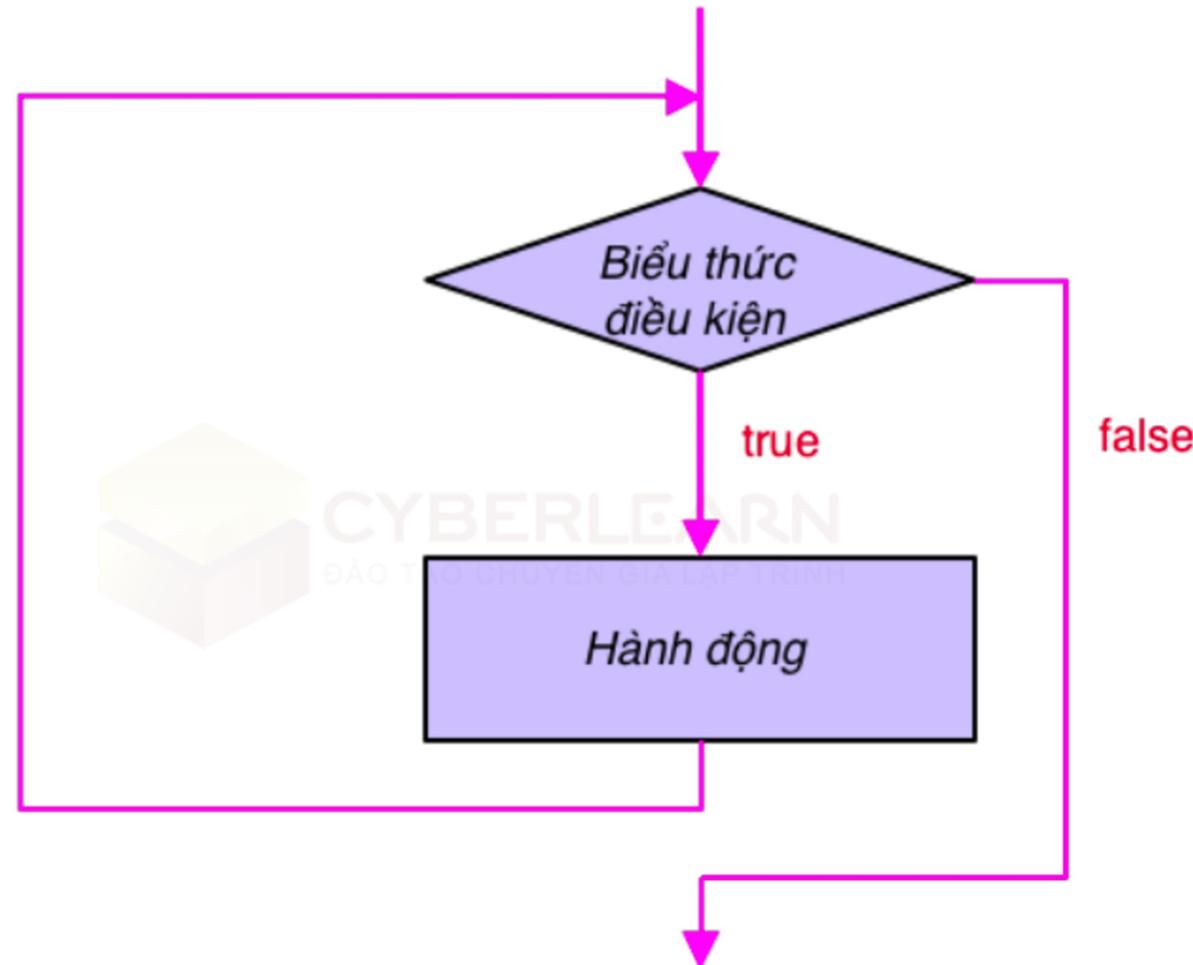
{

CYBERLEARN  
// các dòng lệnh xử lý

// tăng hoặc giảm biến bước nhảy

}

## Lưu đồ hoạt động của vòng lặp while



# Bài tập 1

Viết chương trình, Nếu điểm thi của bạn <5, bạn cần phải thi lại ? Có bao nhiêu lần thi lại?

```
public class WhileDemo {  
    public static void main(String[] args)  
    {  
        float diem = 2;  
        while(diem<5){  
            System.out.println("Điểm lần này của bạn bao nhiêu?");  
            Scanner scan = new Scanner(System.in);  
            diem = scan.nextFloat();  
        }  
        System.out.println("Xin chúc mừng bạn đã ĐẬU");  
    }  
}
```



output

```
> Điểm lần này của bạn bao nhiêu? 4  
> Điểm lần này của bạn bao nhiêu? 2  
> Điểm lần này của bạn bao nhiêu? 3  
> Điểm lần này của bạn bao nhiêu? 7  
  
> CHÚC MỪNG BẠN ĐÃ ĐẬU
```

## Bài tập 2

Tính tổng các số lớn hơn không do người dùng nhập vào.  
Chương trình sẽ dừng khi nhập vào số bằng 0 hoặc số âm.

## Cú pháp vòng lặp do...while

do

{

// các dòng lệnh xử lý

}

while (điều kiện);

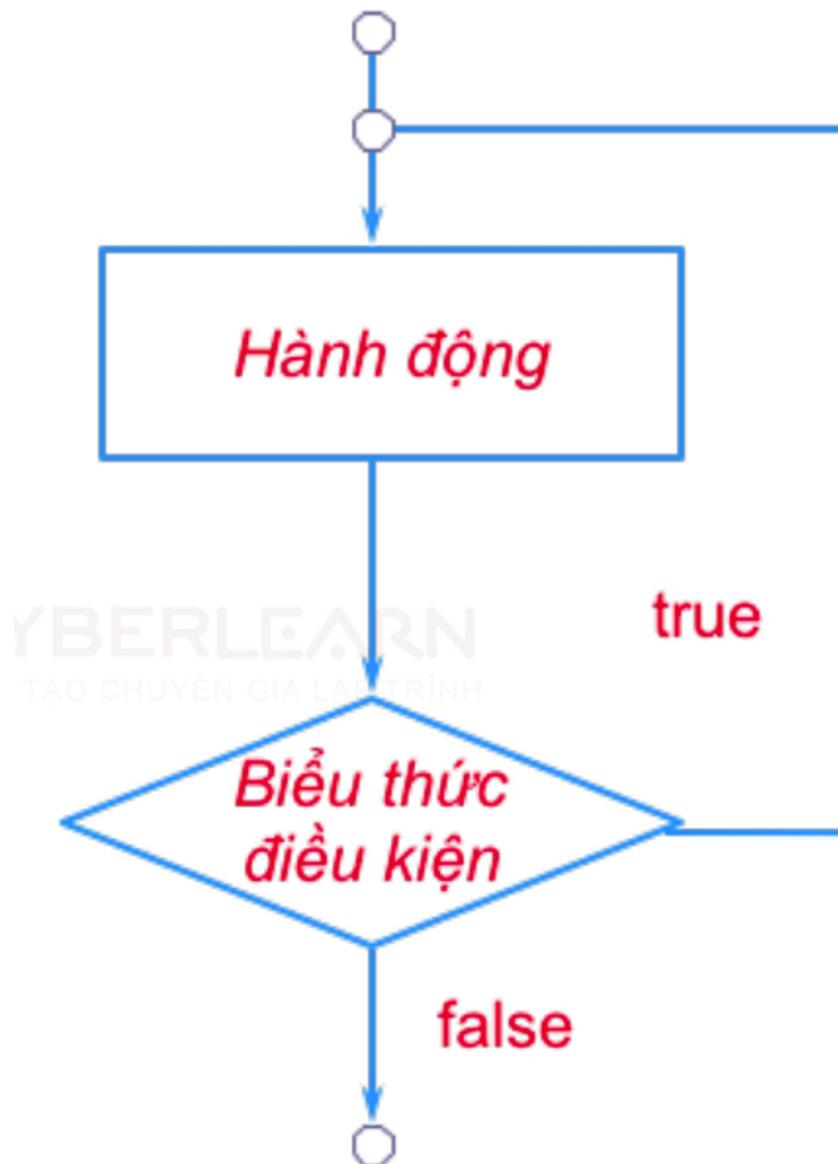
## Cách hoạt động của vòng lặp do...while

```
do
{
    // các dòng lệnh xử lý
}
while (điều kiện); ✓
```

## Cách hoạt động của vòng lặp do...while

```
do  
{  
    // các dòng lệnh xử lý  
}  
while (điều kiện); 
```

## Lưu đồ hoạt động của vòng lặp do...while



## Bài tập 3

Xây dựng hàm ThongBaoNhap và hàm XuLyMenu cho bài tập các phép tính đơn giản với yêu cầu như sau:

Thông báo cho người dùng chọn :

- 1 : Tính tổng,
- 2 : Tính hiệu,
- 3 : Tính nhân,
- 4: Tính chia ,
- 0 : Thoát.

Tất cả nhập khác đều không hợp lệ.

# Trò chơi đoán số

Chương trình cho phép máy tính phát sinh ngẫu nhiên một số từ 1 đến 1000. Người dùng sẽ đoán và nhập, nếu đúng với kết quả của máy, game sẽ dừng. Ngược lại máy sẽ báo bạn đang nhập một số lớn hơn hoặc nhỏ hơn số bí mật.

Hàm tạo số ngẫu nhiên :  
(int) (Math.random() \* 999 + 1);

# Khác nhau giữa while và do...while

Nếu điều kiện là false:

- Lệnh while sẽ không thực thi các lệnh bên trong
- Lệnh do-while sẽ thực thi 1 lần



```
while(false) {  
    //Không được gọi  
    System.out.println ("xin chao");  
}  
  
do {  
    //chạy 1 lần  
    System.out.println ("xin chao");  
} while(false)
```

## Vòng lặp For

Cho biết trước số lần lặp lại

## Cú pháp vòng lặp For

```
for ( khởi tạo; điều kiện; bước nhảy )
{
    // các dòng lệnh xử lý
}
```

## Cách hoạt động của vòng lặp For

Bước lặp

1

```
for ( khởi tạo ; điều kiện; bước nhảy )
{
    // các dòng lệnh xử lý
}
```

CYBERLEARN  
BẢN TÀI ĐIỀU HUYỆN GIÁ LẬP TRÌNH

## Cách hoạt động của vòng lặp For

Bước lặp

2

```
for ( khởi tạo ; điều kiện; bước nhảy )
{
    // các dòng lệnh xử lý
}
```

1



3

2

CYBERLEARN  
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

## Cách hoạt động của vòng lặp For

Bước lặp

3

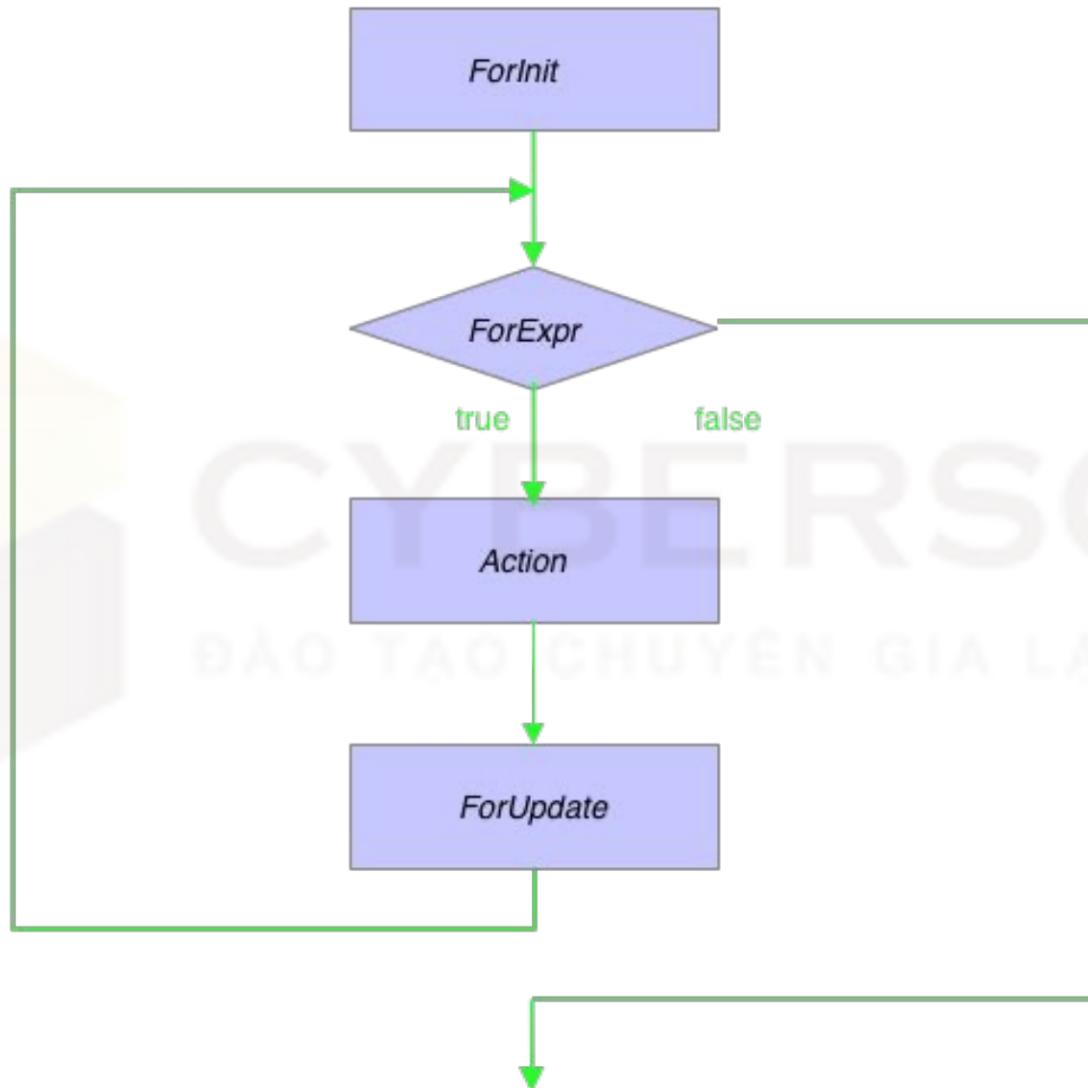
```
for ( khởi tạo ; điều kiện; bước nhảy )
{
    // các dòng lệnh xử lý
}
```

1



CYBERLEARN  
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

# Vòng lặp For



# MỘT SỐ GHI CHÚ

```
for ( khởi tạo ; điều kiện; bước nhảy )  
{  
    // các dòng lệnh xử lý  
}
```

**Khởi tạo:**  
\* Bắt đầu từ 0, 1, ...

**Bước nhảy:**  
\* ++, +=2 , \*=3, --, -=2

## Luyện tập vòng lặp For

- 1. In tất cả số nguyên dương lẻ / chẵn nhỏ hơn 100. (Viết cả while và for) -  
Dùng bước nhảy và dùng % 2**
- 2. Tính tổng các số chẵn từ 1 đến n. Với n nhập từ người dùng (While và for)**
- 3. Có bao nhiêu số chia hết cho 3 từ 0-1000 (While và for)**



# Lệnh Break và Continue

- Break: Dừng vòng lặp ngay tại thời điểm gọi
- Continue : Nhảy đến bước nhảy tiếp theo. (Không nên lạm dụng lệnh này vì sẽ gây rối chương trình)

```
while (<điều kiện>)
```

```
{
```

```
    if(<điều kiện>) {
```

```
        // các lệnh xử lý
```

```
        break;
```



```
}
```

```
        // các lệnh xử lý
```

```
}
```

```
    // các dòng lệnh xử lý tiếp theo
```

Vòng lặp ngoài

```
while ( <điều kiện> )
{
    Vòng lặp trong
    while (<điều kiện> )
    {
        if(<điều kiện>)
            break;
    }
    // Các lệnh xử lý
}
```

## Lệnh Continue

```
for(khởi tạo; điều kiện; bước nhảy){  
    if(điều kiện){  
        //Các lệnh xử lý  
        continue;  
    }  
    //Các lệnh xử lý  
}
```

CYBERSOFT  
TRUNG TÂM CHUYÊN GIA LẬP TRÌNH

# Bài tập

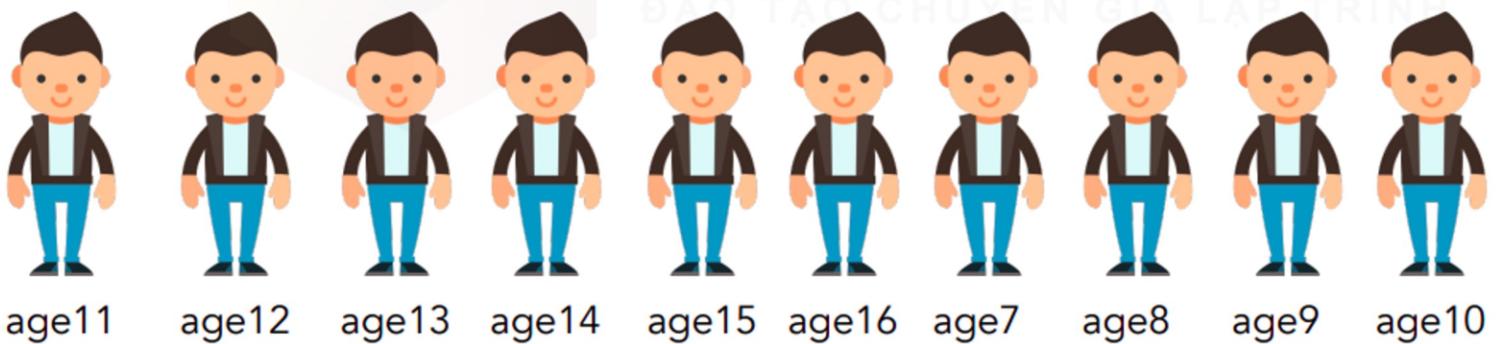
1. Xây dựng chương trình trò chơi Oẳn Tù Tì (Bao, kéo, kim). Người và máy sẽ chơi với nhau. Nhấn một số khác 0 để chơi tiếp, nhấn 0 để dừng. Tính xem ai sẽ là người thắng nhiều nhất và tỉ số là bao nhiêu.
2. Tìm số nguyên dương n nhỏ nhất sao cho :  $1 + 2 + \dots + n > 10000$
3. Viết chương trình tính tổng các số lẻ nguyên dương nhỏ hơn n được nhập từ người dùng. Chỉ cho nhập số  $> 0$ , yêu cầu nhập lại nếu không đúng yêu cầu.
4. Viết chương trình nhập n và tính tổng  
 $S(n) = x + x^2 + x^3 + \dots + x^n$  (Sử dụng vòng lặp + hàm)
5. Chương trình tính và in số năm chờ đợi. Anh X có một số tiền nhàn rỗi. Anh đem gởi tiết kiệm số tiền này. Anh X hy vọng rằng đến một lúc nào đó trong tương lai anh sẽ có đủ tiền để mua một chiếc ô tô. Cho biết số tiền anh X gửi, số tiền anh X muốn có trong tương lai và tiền lãi tiết kiệm. Hãy viết chương trình tính số năm ít nhất mà anh X phải chờ đợi kể từ năm gởi tiền. Giả sử lãi suất tiết kiệm không đổi, anh X không rút bớt tiền và cũng không gởi thêm tiền trong thời gian chờ đợi.

# Mảng - Vòng lặp

- Tại sao có mảng ?
- Lưu trữ 100 số ? 1000 sinh viên ?
- Ví dụ về mảng

CYBERSOFT  
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

# GIỚI THIỆU MẢNG (ARRAY)

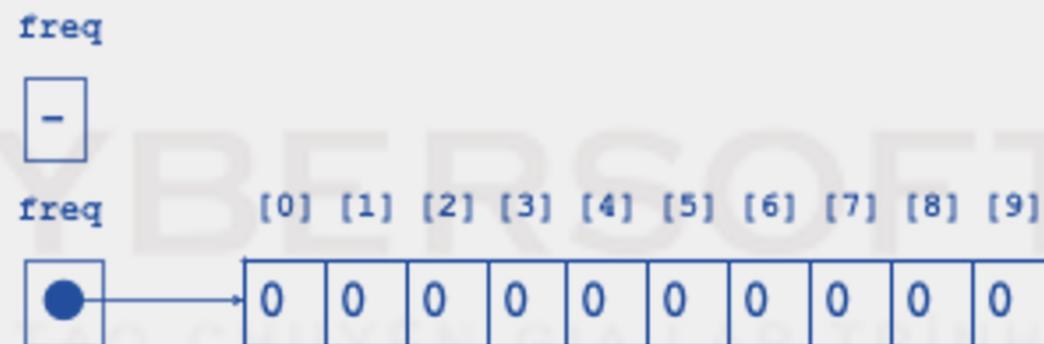


# GIỚI THIỆU MẢNG (ARRAY)

1		Drew Vosburg	9,735
2		Jesse Nelson	8,364
3		Joel Bullis	7,621
4		Carmen SanDiego	4,582
5		Charles Barkley	2,651

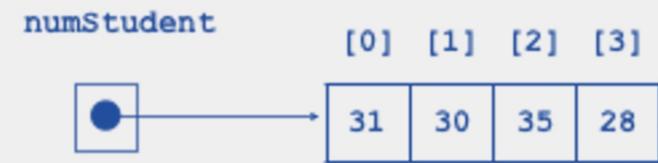
# Mảng - bản chất lưu trữ

```
int [] freq;  
  
freq = new int[10];
```

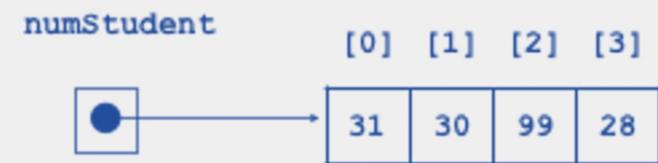


# Mảng - bản chất lưu trữ

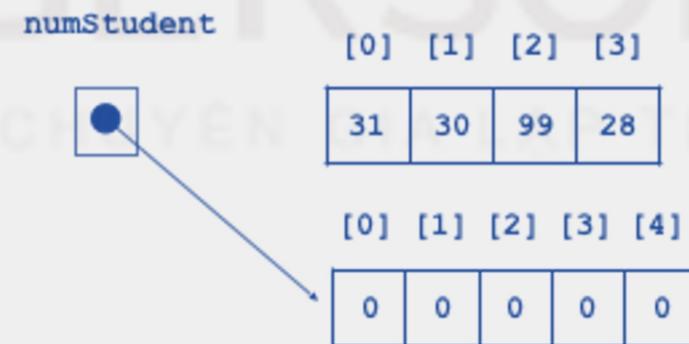
```
int [] numStudents = {31,30,35,28};
```



```
numStudents[2] = 99;
```



```
numStudents = new int[5];
```



# KHAI BÁO MẢNG (ARRAY)

Kiểu dữ liệu [ ]

Tên mảng

hoặc

Kiểu dữ liệu

Tên mảng [ ]

```
//Khai báo mảng kiểu int  
int[] arr1;  
//Khai báo mảng kiểu long  
long[] arr2;  
//Khai báo mảng kiểu float  
float[] arr3;  
//Khai báo mảng kiểu double  
double[] arr4;  
//Khai báo mảng kiểu boolean  
boolean[] arr5;  
//Khai báo mảng kiểu string  
String[] arr6
```

# Cấp phát vùng nhớ

Cách 1	KieuDuLieu[] tenBien = new KieuDuLieu [n] ;
Cách 2	KieuDuLieu[] tenBien; tenBien = new KieuDuLieu [n] ;

```
1  
2 //Khai báo và cấp phát mảng kiểu int  
3 int[] arr1 = new int[5];// a.length = 5  
4 //Khai báo và cấp phát mảng kiểu long  
5 long[] arr2 = new long[5];// a.length = 5  
6 //Khai báo và cấp phát mảng kiểu float  
7 float[] arr3 = new float[7];//a.length = 7  
8 //Khai báo và cấp phát mảng kiểu double  
9 double[] arr4 = new double[7];//a.length = 7  
10 //Khai báo và cấp phát mảng kiểu boolean  
11 boolean[] arr5 = new boolean[8];//a.length = 8  
12 //Khai báo và cấp phát mảng kiểu string  
13 String[] arr6 = new String[6];//a.length = 6
```

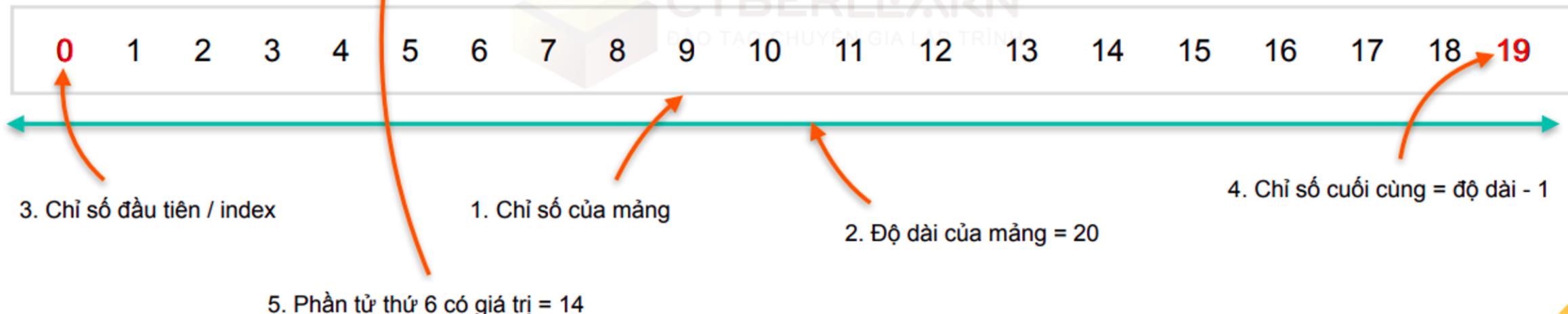
# Khởi tạo

```
1 //Cách 1
2 int[] arr = {1, 3, 5, 7, 9}; //a.length = 5
3 //Cách 2
4 int [] arr = new int[5];
5 arr[0]=1;
6 arr[1]=3;
7 arr[2]=5;
8 arr[3]=7;
9 arr[4]=9;
```

# CÁC PHẦN LIÊN QUAN ĐẾN MẢNG (ARRAY)

int [ ] age

age[0] age[1] age[2] age[3] age[4] age[5] age[6] age[7] age[8] age[9] age[10] age[11] age[12] age[13] age[14] age[15] age[16] age[17] age[18] age[19] age[20]  
[ ] [ ] [ ] [ ] [ ] [14] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ]



# Câu lệnh foreach

```
//tenBien lần lượt là các phần tử bên trong mảng  
for (KieuDuLieu tenBien : tenMang) {  
    Các câu lệnh;  
}
```

```
1 System.out.println("Xuất mảng dùng foreach");  
2 System.out.println("Số phần tử của mảng " + a.length);  
3 for (int pt : a){  
4     System.out.println(pt);  
5 }
```

```
1 System.out.println("Xuất mảng dùng for bình thường");  
2 System.out.println("Số phần tử của mảng " + a.length);  
3 for (int i=0 ; i<a.length; i++){  
4     System.out.println(a[i]);  
5 }
```

# Nhập và xuất mảng một chiều

```
1 //Nhập mảng
2 System.out.print("Số phần tử của mảng là ");
3 int n = Integer.parseInt(scan.nextLine());
4 int [] a = new int [n]; //a.Length = n
5 for (int i = 0; i < a.length; i++) {
6     System.out.print("a["+i+"]=");
7     a[i] = Integer.parseInt(scan.nextLine());
8 }
9 //Xuất mảng dùng for
10 System.out.println("Xuất mảng dùng for");
11 System.out.println("Số phần tử của mảng " + a.length);
12 for (int i = 0; i < a.length; i++)
13 {
14     System.out.println(a[i]);
15 }
16 //Xuất mảng dùng foreach
17 System.out.println("Xuất mảng dùng foreach");
18 System.out.println("Số phần tử của mảng " + a.length);
19 for (int pt : a)
20 {
21     System.out.println(pt);
22 }
```

# Tính tổng các phần tử trong mảng một chiều

```
1 //Tính tổng dùng for
2 int s = 0;
3 for (int i = 0; i < a.length; i++)
4 {
5     s = s + a[i];
6 }
7 System.out.println("s = {0}", s);
8 //Tính tổng dùng foreach
9 s = 0;
10 for (int pt : a)
11 {
12     s = s + pt;
13 }
14 System.out.println("s = " + s);
```

# Bài tập

Cho người dùng nhập vào n số nguyên và lưu trữ trong mảng

- a. Viết hàm nhập, hàm xuất
- b. Tìm số lớn nhất và vị trí của nó trong mảng
- c. Tìm số âm đầu tiên và vị trí của nó trong mảng.
- d. Tìm số âm lớn nhất và vị trí của nó trong mảng
- e. Tính tổng các số chẵn.
- f. Đếm có bao nhiêu số âm.
- g. Tổng các số âm.
- h. Cho người dùng nhập vào x và tìm x có trong mảng đó hay không.

# Bài tập

1. Cho người dùng nhập vào n số nguyên và lưu trữ trong mảng
  - a. Làm menu cho người dùng chọn thực hiện các chức năng bên dưới.
  - b. Tổng các số dương.
  - c. Tính tổng các số lẻ.
  - d. Đếm có bao nhiêu số dương.
  - e. Tìm số nhỏ nhất trong mảng.
  - f. Tìm số dương nhỏ nhất trong mảng
  - g. Tìm số chẵn cuối cùng trong mảng 1 chiều các số nguyên. Nếu mảng không có giá trị chẵn thì trả về -1
  - h. Viết hàm tìm số chẵn đầu tiên trong mảng các số nguyên. Nếu mảng không có giá trị chẵn thì trả về -1
  - i. Tìm số nguyên tố đầu tiên trong mảng 1 chiều các số nguyên. Nếu mảng không có số nguyên tố thì trả về – 1
  - j. Tìm số dương cuối cùng trong mảng số thực. Nếu mảng không có giá trị dương thì trả về -1
  - k. Hãy tìm giá trị chẵn nhỏ nhất trong mảng 1 chiều các số nguyên. Nếu mảng không có số chẵn thì trả về -1

## Bài tập

2. Hãy tìm giá trị và vị trí trong mảng số thực “xa giá trị x nhất”. X được nhập từ người dùng.

24	45	23	13	43	-12
----	----	----	----	----	-----

Giá trị x: 15

Khoảng cách từ  $x = 15$  đến các phần tử khác trong mảng là:

9	30	8	2	28	27
---	----	---	---	----	----

Giá trị trong mảng xa nhất là **45** tại vị trí thứ **1**

# Bài tập

3. Cho mảng 1 chiều các số thực, hãy tìm đoạn  $[a, b]$  sao cho đoạn này chứa tất cả các giá trị trong mảng
4. Cho mảng 1 chiều các số thực, hãy tìm giá trị đầu tiên lớn hơn giá trị 2018.  
Nếu mảng không có giá trị thỏa điều kiện trên thì trả về -1
5. Cho mảng 1 chiều các số thực, hãy tìm giá trị âm cuối cùng lớn hơn giá trị -1.  
Nếu mảng không có giá trị thỏa điều kiện trên thì trả về -1
6. Hãy liệt kê các số âm trong mảng 1 chiều các số thực