

**Mảng**

*Cấp phát bộ nhớ động*

\*ptr = new kieu\_du\_lieu[so\_phan\_tu];

Ex: int \*arr;

arr = new int[100];

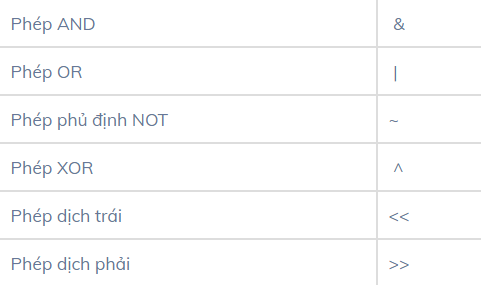
*Giải phóng bộ nhớ*

delete[] ten\_mang;

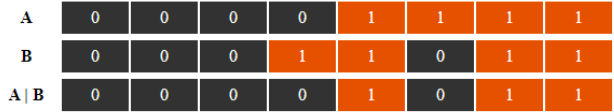
**Các phép toán thao tác bit**

Việc biết về các phép toán trên bit có thể giúp bạn viết chương trình chạy nhanh hơn

Ex: Câu lệnh if (a & 1) sẽ có tốc độ thực thi nhanh hơn if (a % 2 != 0) do đây là phép toán trên bit.



Phép AND giữa 2 bit sẽ trả về 1 khi cả 2 bit đều có giá trị bằng 1, ngược lại nó sẽ trả về 0



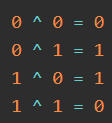
* Ứng dụng

Check even (odd) number => if (a & 1)

Phép OR giữa 2 bit sẽ trả về 1 nếu có ít nhất 1 trong 2 bit có giá trị bằng 1, ngược lại nó sẽ trả về 0

Phép NOT là phép đảo bit

Phép XOR giữa 2 bit sẽ trả về 1 khi 2 bit có giá trị khác nhau, ngược lại nó sẽ trả về 0



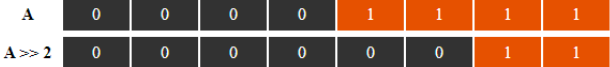
Phép dịch trái là phép dịch tất cả các bit sang trái



* Ứng dụng

Tính 2 ^ n => 1 << n

Phép dịch phải là phép dịch tất cả các bit sang phải



* Ứng dụng

Lấy phần nguyên của phép chia x / 2; (x là giá trị đầu vào) => x >> 1

Chú ý:

* Khi sử dụng phép dịch phải (trái) cần kết hợp với phép gán

Ex: a = a >> 1; b = b << 2

**Kiểu cấu trúc**

trong C++ có các kiểu dữ liệu có sẵn như string, int, short, long long, ... >> struct

* Thuộc tính

#include<iostream>

using namespace std;

// khai bao kieu du lieu struct voi 2 thuoc tinh name va age

struct Student

{

    string name;

    int age;

};

int main() {

    // tao ra 1 doi tuong kieu Student de luu thong tin cua 1 sinh vien

    Student s;

    cin >> s.name >> s.age;

    cout << "Name: " << s.name << endl << "Age: " << s.age;

    return 0;

}

Student s = students[0];

for (int i = 1; i < n; i++) {

if (students[i].score > s.score) {

s = students[i];

}

}

cout << "Name: " << s.name << endl;

cout << "Age: " << s.age << endl;

cout << "Score: " << s.score;

* Phương thức
* Định nghĩa các phương thức bên trong 1 struct

Ex: sử dụng phương thức display() >> hiển thị thông tin của 1 sinh viên

struct Student {

string name;

int dateOfBirth;

void display() {

cout << "Name: " << name << endl;

cout << "Age: " << dateOfBirth;

}

* Tính chất

Phương thức có thể truy xuất tới thuộc tính của sturct

Ex: phương thức getInformation()

struct Student {

    string name;

    int age;

    void display() {

        cout << "Name: " << name << endl << "Age: " << age;

    }

    void getInformation() {

        cin >> name >> age;

    }

};