

Tin học cơ sở 4

Buổi 7. Nhập xuất tệp

Bộ môn Khoa học máy tính - 2017

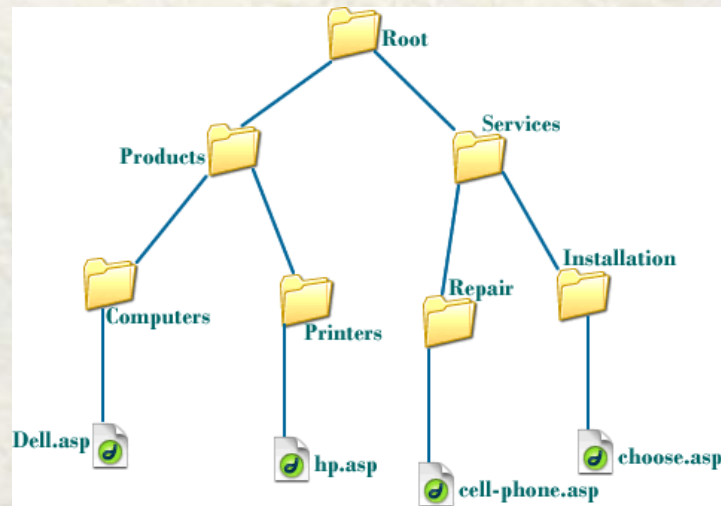


Nội dung buổi học

1. Sử dụng luồng nhập (từ tệp)
2. Sử dụng luồng xuất (ra tệp)
3. Thao tác với tệp
 - Duyệt thư mục
 - Sao chép tệp
 - Xóa tệp

Cấu trúc thư mục

- Một ổ đĩa gồm nhiều thư mục và tệp tin
- Mỗi thư mục lại gồm nhiều thư mục con và tệp tin khác

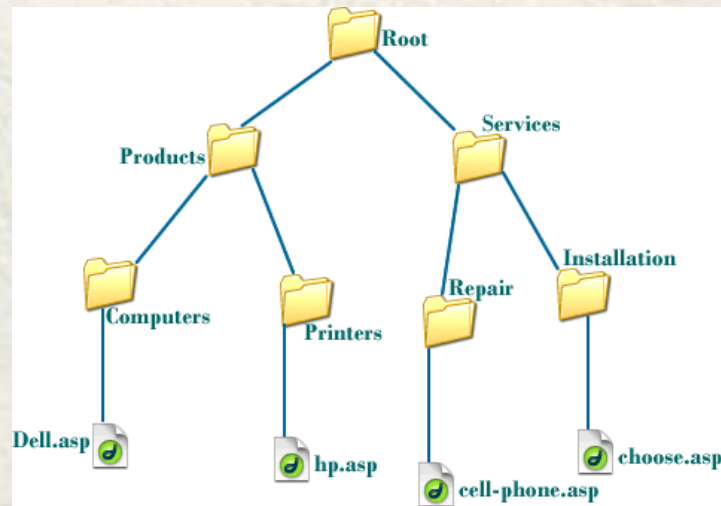


Cấu trúc thư mục

- Hệ điều hành cung cấp các chương trình duyệt cấu trúc thư mục và đọc các tệp tin

Tự lập trình ?

HOUR
OF
CODE



Explorer



Nautilus



Tập tin

- Tập hợp thông tin được đặt tên
 - ❖ alarm.wav, input.txt, ...
- Chứa trong các thiết bị lưu trữ
 - ❖ Đĩa cứng, đĩa mềm, USB flash, CD, DVD
- Còn gọi là tệp tin, tệp hoặc file
- Đơn vị nhỏ nhất của tệp là **byte**

Đọc tệp văn bản

➤ C++ cung cấp kiểu `ifstream`

❖ Dùng để đọc tệp văn bản

❖ `#include <fstream>`

❖ Cách dùng rất giống `cin`

Toán tử `>>`

Lệnh `getline`

❖ Tham khảo tại

<http://www.cplusplus.com/reference/fstream/ifstream/>

Đọc tệp văn bản

- Khai báo và mở tệp để đọc
`ifstream <tên biến>(<xâu chứa
tên file>);`

Ví dụ:

```
ifstream inFile("paragraph.txt");
```

- Nếu thành công, tệp sẽ được mở ra
- Có thể bắt đầu đọc từ đầu tệp

Kiểm tra lỗi

- Kiểm tra thao tác với file có lỗi không

```
if (inFile) {
```

```
    // code nếu thao tác tệp thành công
```

```
} else {
```

```
    // code nếu thao tác tệp thất bại
```

```
}
```

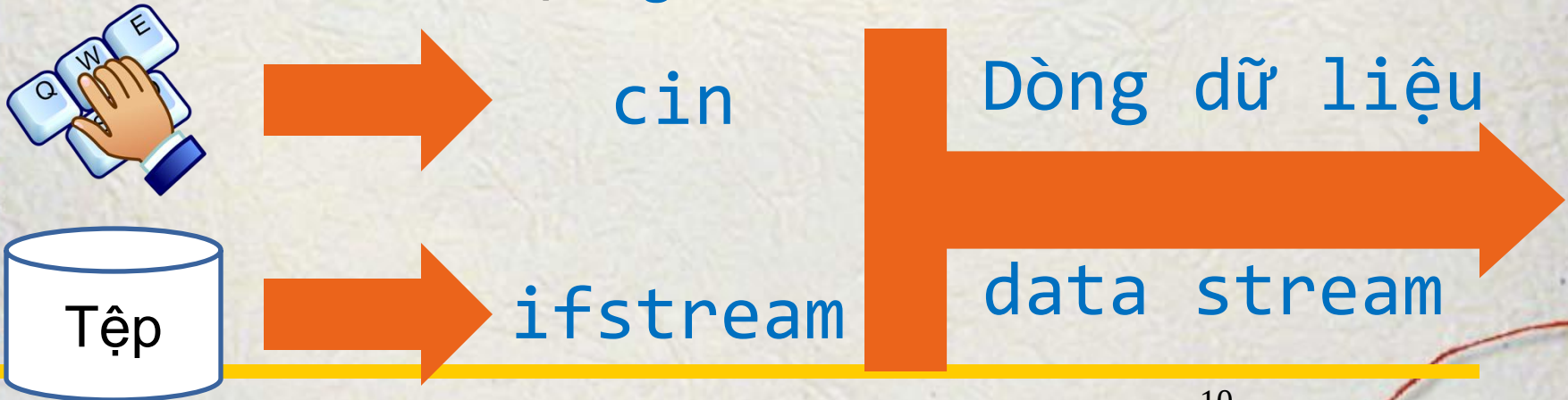
- *Luôn luôn kiểm tra sau mỗi lần thao tác với tệp*

Kiểm tra lỗi

```
ifstream inFile("paragraph.txt");  
if (inFile) {  
    cout << "File is ready to read\n";  
} else {  
    cout << "Error: cannot open file\n";  
}
```

Luồng dữ liệu vào (istream)

- `cin` là luồng dữ liệu từ luồng nhập chuẩn
 - ❖ ví dụ: bàn phím
- `ifstream` là luồng dữ liệu từ tệp
- Sau khi mở tệp, thao tác giống nhau
 - ❖ Toán tử `>>`, lệnh `getline` ...



Đọc dữ liệu từ **ifstream**

➤ Sử dụng toán tử **>>** đọc số, ký tự

Ví dụ:

```
int age;
```

```
double angle;
```

```
string str;
```

```
inFile >> age >> angle >> str;
```

Các giá trị cách nhau bằng khoảng trắng

Đọc dữ liệu từ **ifstream**

- Sử dụng lệnh **getline** đọc một dòng

Ví dụ:

```
string line;  
getline(inFile, line);
```

Đọc dữ liệu từ **ifstream**

- **Lưu ý:** khi đọc cả số lẫn dòng
 - ❖ Cần thêm một lệnh **getline**

```
int age;  
string line;  
inFile >> age;  
// phần còn lại của dòng chứa giá trị age  
getline(inFile, line);  
// dòng tiếp theo sau giá trị age  
getline(inFile, line);
```

Sử dụng giá trị này

Đọc nội dung tệp văn bản

```
int main()
{
    ifstream file("file.cpp");
    if (!file) {
        cout << "Error: cannot open file\n";
        return 0;
    }
    int count = 1;
    while (!file.eof()) {
        string line;
        getline(file, line);
        if (file) {
            cout << count << ": " << line << endl;
            count++;
        }
    }
    return 0;
}
```

Kiểm tra lỗi mở tệp

Kiểm tra có hết tệp ?

Kiểm tra lỗi đọc dòng

In dòng nếu không có lỗi

Đọc nội dung tệp văn bản

```
int main()
```

```
{
```

```
    ifstream file("number.txt");
```

```
    if (!file) {
```

```
        cout << "Error: cannot open file\n";
```

```
        return 0;
```

```
    }
```

```
    while (!file.eof()) {
```

```
        int value;
```

```
        file >> value;
```

```
        if (file)
```

```
            cout << value << " ";
```

```
    }
```

```
    cout << endl;
```

```
    return 0;
```

```
}
```

Kiểm tra lỗi mở tệp

Kiểm tra có hết tệp ?

Đọc số bằng toán tử >>

Kiểm tra lỗi đọc số

In số nếu không có lỗi

Ghi ra tệp văn bản

➤ C++ cung cấp kiểu `ofstream`

❖ Dùng để ghi ra tệp văn bản

❖ `#include <fstream>`

❖ Cách dùng rất giống `cout`
Toán tử `<<`

❖ Tham khảo tại

<http://www.cplusplus.com/reference/fstream/ofstream/>

Luồng dữ liệu ra (ostream)

- **cout** là luồng dữ liệu ra luồng xuất chuẩn
 - ❖ ví dụ: màn hình văn bản
- **ofstream** là luồng dữ liệu ra tệp
- Sau khi mở tệp, thao tác giống nhau
 - ❖ Toán tử << ...



Mở tệp văn bản để ghi

- Khai báo và mở tệp để ghi
`ofstream <tên biến>(<xâu chứa tên file>);`

Ví dụ:

```
ofstream outFile("paragraph.txt");
```

- Nếu thành công, tệp sẽ được mở ra
- Có thể bắt đầu ghi thông tin vào tệp

Ghi dữ liệu vào **ofstream**

➤ Sử dụng toán tử << để ghi số, ký tự

Ví dụ:

```
int age = 5;
```

```
double angle = 0.6;
```

```
string str = "Hello, World!";
```

```
outFile << age << angle << str;
```

Đọc số, sắp xếp và ghi ra tệp

```
int main()
{
    ifstream file("number1.txt");

    double a[100];
    int n = 0;

    while (file && !file.eof()) {
        double value;
        file >> value;
        if (file) {
            a[n] = value;
            n++;
        }
    }
}
```

```
ofstream fileOut("number1.out");
sort(a, a+n);
for (int i = 0; i < n; i++)
    fileOut << a[i] << " ";
fileOut << endl;

return 0;
}
```

Sắp xếp
<algorithm>

Đọc dãy số, lưu vào mảng biết số lượng số

```
int main()
{
    ifstream file("number.txt");

    int a[100];
    int n;

    file >> n;
    for (int i = 0; i < n; i++)
        file >> a[i];

    // in mảng ...

    return 0;
}
```

Đọc vào số lượng số
trước khi lặp



Đọc dãy số, lưu vào mảng chưa biết số lượng số

```
int main()
{
    ifstream file("number1.txt");
    double a[100];
    int n = 0;
    while (file && !file.eof()) {
        double value;
        file >> value;
        if (file) {
            a[n] = value;
            n++;
        }
    }

    return 0;
}
```

Khởi tạo số lượng số ban đầu bằng 0

Nếu đọc số thành công

Ghi số đó vào a[n]

Tăng số lượng số lên 1