

Bài thực hành:

Lập trình với nhập xuất tập tin văn bản

A. Mục tiêu

- Giúp sinh viên biết cách thao tác trên file văn bản
- Cung cấp sinh viên kiến thức về đọc và ghi file văn bản
- Kết quả sau khi sinh viên thực hiện bài lab: Đọc và ghi file:
 - Mảng 1 chiều
 - Ma trận
 - Mảng cấu trúc

B. Yêu cầu thực hành

- Sinh viên hiểu được các thao tác đọc và ghi file văn bản
- Thực hiện được bài thực hành về đọc và ghi file văn bản
 - Mảng 1 chiều
 - Ma trận
 - Mảng cấu trúc

C. Hướng dẫn lý thuyết

1. Khai báo thư viện sử dụng <fstream>

2. Sử dụng luồng ghi/đọc:

- Đối tượng
 - `fstream` //Luồng đọc và ghi
 - `ofstream` // Luồng ghi
 - `ifstream` //Luồng đọc

3. Tạo luồng

- `fstream Tên_Luong; //Nhập /Xuất.`
- `ifstream Tên_Luong; //Nhập .`
- `ofstream Tên_Luong; //Xuất.`

4. _Mở tập tin (đọc – nhập, ghi-xuất)

a. Mở để đọc:

```
fstream in; //Tạo luồng nhập, xuất
in.open("test", ios::in, 0);
```

Hoặc:

```
ifstream in; //Tạo luồng nhập
in.open("test");
```

b. Mở để ghi: Các cách mở sau là tương đương: Khi dùng ofstream

```
fstream out; //Tạo luồng nhập, xuất
out.open("test", ios::out, 0);
```

Hoặc:

```
ofstream out; //Tạo luồng xuất
out.open("test");
```

c. Mở tập tin nhập/xuất: dùng fstream

```
fstream mystream;
mystream.open("test", ios::in | ios::out); //Kết hợp in,out
```

d. Kết hợp tạo luồng vừa mở luôn tập tin.

Khởi tạo một đối tượng fstream ngay khi nó được khai báo.

- Đọc:
 fstream in ("test",ios::in);
hoặc
 ifstream in("test")
- Ghi:
 fstream out ("test",ios::out);
hoặc
 ofstream out("test");

Tóm lại, các cách viết sau là tương đương:

fstream in; in.open("test",ios::in);	fstream in ("test",ios::in);	ifstream in("test")
fstream out; out.open("test",ios::out);	fstream out ("test",ios::out);	ofstream out("test")

5. *Kết quả của việc mở tập tin tin.*

Nếu việc mở tập tin (đọc hay ghi, gắn với luồng đã mở) không thành công, luồng sẽ trả về giá trị 0

6. Kiểm tra mở tập tin có thành công hay không:

Xem khai báo:

```
fstream mystream;
mystream.open("test", ios:: in | ios:: out);
```

Cách 1:

Kiểm tra giá trị của luồng (Giá trị luồng trả về 0 nếu không mở được tập tin)

```
if ( !mystream)
{
    cout<<"\nTập tin không mở được!";
    exit(1);
}
```

Cách 2:

Dùng hàm thành phần fail(). Hàm trả về giá trị 1 nếu việc mở tập tin không thành công.

```
if (mystream.fail())
{
    cout<< "\nTập tin không mở được!";
    exit(1);
}
```

7. Kiểm tra hết tập tin.

Dùng hàm thành phần eof() của luồng ?fstream

$$?fstreamName.eof() = \begin{cases} 1; & \text{Hết dữ liệu} \\ 0; & \text{Ngược lại} \end{cases}$$

trong đó ? là **i, o** hoặc **không có ký tự nào**

Với khai báo:

```
ifstream in("test");
if(!in.eof()) //chưa hết dữ liệu
{
    //Xử lý dữ liệu
}
```

8. *Thao tác đọc dữ liệu từ tập tin, ghi dữ liệu vào tập tin.*

Với các khai báo:

```
ifstream in("test1");
ofstream out("test2");
int sonhap;
```

- Đọc:

```
in>>sonhap; // đọc dữ liệu trong test 1, lưu trữ trong sonhap
```

- Ghi:

```
out<<sonhap; //ghi sonhap vào test2
```

9. *Đóng tập tin:*

Dùng hàm thành phần close()

```
mystream.close ();
```

Lưu ý: Đối với tập tin, đã có thao tác mở thì nên có thao tác đóng.

D. Hướng dẫn thực hành

Bài 1: (Chuyển dữ liệu của Tập tin vào Mảng 1 chiều)

Đọc dữ liệu của tập tin văn bản – ghi vào mảng một chiều

Giả sử tập tin văn bản \test1.txt có nội dung sau:

Tập tin:

```
9
1    2    3    4    5    6    7    8    9
```

Trong đó:

- Hàng 1: chứa số lượng các phần tử của tập tin (9)
- Hàng 2: Các giá trị của tập tin.

Viết chương trình đọc các giá trị trong tập tin \test1.txt rồi lưu trữ vào mảng 1 chiều các số nguyên.

```
#include <iostream>
#include <fstream>
#include <conio.h>
#define MAX 100
```

```

using namespace std;
void File_Array(char *filename, int Arr[MAX], int &n);
int main()
{
    int n, Arr[MAX];
    char filename[MAX];
    system("cls");
    cout<<"Nhập tên file mô đề doc:";
    cin>>filename;
    File_Array(filename,Arr,n);
    cout<<endl;
    cout<<n<<endl;
    for (int i = 0; i < n; i++)
        cout<<Arr[i]<<"\t";
    system("PAUSE");
    return 0;
}
void File_Array(char *filename, int Arr[MAX], int &n)
{
    ifstream in(filename);    //Mở tập tin filename để đọc
    if (!in) //Kiểm tra việc mở tập tin
    {
        cout<<"\nLỗi mở file !";
        exit(1);
    }
    in>>n; //đọc dữ liệu đầu tiên của tập tin (hàng đầu), xác định kích thước mảng
    for (int i = 0; i < n; i++)
    {
        in>>Arr[i]; // lần lượt đọc dữ liệu tập tin ghi vào mảng 1 chiều
    }
    in.close();
}

```

Bài 2: (Chuyển dữ liệu của Mảng 1 chiều vào Tập tin)

Đọc dữ liệu của mảng 1 chiều rồi ghi vào tập tin “\Test.txt” theo định dạng :

- Hàng đầu : số phần tử của tập tin (kích thước của mảng)
- Các hàng sau là các phần tử của tập tin (các phần tử của mảng)

Mảng a :

```
int a[9] = {1,2,3,4,5,6,7,8,9};
```

Tập tin :

```

9
1      2      3      4      5      6      7      8      9

```

Viết hàm thực hiện thao tác trên và có kiểm tra lại bằng chương trình.

```

#include <iostream>
#include <fstream>
#include <conio.h>
using namespace std;
void write_int(int a[], int n, char *filename);
int main()
{
    int n = 9;
    int a[] = {1,2,3,4,5,6,7,8,9};

    char filename[80];
    system("cls");
    cout<<"Nhập tên file mô đề ghi:";cin>>filename;
    write_int(a,n,filename);
    system("PAUSE");
    return 0;
}
void write_int(int a[], int n, char *filename)
{
    ofstream out(filename);    //Mô đề ghi
    if (!out)
    {
        cout<<"\nLỗi mô đề ghi!";
        exit(1);
    }
    out<<n;
    for(int i = 0; i < n;i++)
    {
        out<<a[i];
        out<<"\t";
    }
    cout<<"\nghi xong dữ liệu vào tệp "<<filename;

```

```

        out.close();
    }

```

Bài 3 : (Chuyển dữ liệu của Tập tin vào Mảng 1 chiều, định dạng tập tin chỉ chứa dữ liệu – không có số phần tử tập tin)

Giả sử tập tin văn bản \test2.txt có nội dung là các số nguyên sau:

1 2 3 4 5 6 7 8 9

Viết hàm đọc các giá trị trong tập tin \test2.txt rồi lưu trữ vào mảng 1 chiều các số nguyên (hoặc xuất ra màn hình).

```

#include <iostream>
#include <fstream>
#include <conio.h>
#define MAX 100
using namespace std;

void File_Array1(char *filename, int arr[MAX], int &n);

int main()
{
    int n, arr[MAX];
    char filename[80];
    system("cls");
    cout<<"Nhập tên file mô de doc:";cin>>filename;
    File_Array1(filename,arr,n);
    cout<<"\nN="<<n;
    cout<<endl;
    for (int i = 0; i < n; i++)
        cout<<arr[i]<<"\t";

    system("PAUSE");
    return 0;
}

void File_Array1(char *filename, int arr[MAX], int &n)
{
    ifstream in(filename);    //Mô de doc
    if (!in)
    {
        cout<<"\nLỗi mô file !";
        exit(1);
    }
}

```

```

    }
    n = 0;
    in >> arr[n];
    while (!in.eof())
    {
        n++;
        in >> arr[n];
    }
    n++;
    in.close();
}

```

Bài 4: (Chuyển dữ liệu của Tập tin vào Ma trận vuông)

Đọc dữ liệu tập tin (hàng đầu của tập tin là cấp ma trận vuông) rồi ghi vào ma trận vuông)

Giả sử tập tin văn bản \test3.txt có nội dung sau:

```

3
1      2      3
4      5      6
7      8      9

```

Trong đó:

- Hàng 1: Chỉ cỡ của ma trận vuông.
- Các hàng sau là các phần tử của tập tin

Viết hàm đọc các giá trị trong tập tin \test3.txt rồi lưu trữ vào ma trận vuông cấp n.

```

#include <iostream>
#include <fstream>
#include <conio.h>
#define MAX 100
using namespace std;

```

```

void File_Mat(char *filename, int a[MAX][MAX], int &n);

```

```

int main()
{
    int n, a[MAX][MAX], i, j;
    char filename[80];

```



```

system("cls");
cout<<"Nhập tên file mode đọc:";cin>>filename;
File_Mat(filename,a,n);
cout<<"\nN="<<n;
cout<<endl;
for ( i = 0; i < n; i++)
{
    cout<<"\n";
    for (j = 0; j < n; j++)
        cout<<a[i][j]<<"\t";
}
system("PAUSE");
return 0;
}
void File_Mat(char *filename, int a[MAX][MAX], int &n)
{
    ifstream in(filename);    //Mode đọc
    if (!in)
    {
        cout<<"\nLỗi mở file !";
        exit(1);
    }
    in>>n;
    int i, j;
    for ( i = 0; i < n; i++)
        for (j = 0; j < n; j++)
            in>>a[i][j];
    in.close();
}

```

Bài 5: (Chuyển dữ liệu ma trận vuông vào Tập tin theo định dạng)

Duyệt ma trận vuông, ghi dữ liệu của ma trận vào tập tin theo định dạng như sau:

- Dòng 1: n //Cấp ma trận vuông
- Các dòng sau: Ghi các giá trị ma trận theo n dòng và n cột tương ứng. Các giá trị cách nhau 1 tab.

Chẳng hạn:

```

3
1    2    3

```

4	5	6
7	8	9

```

#include <iostream>
#include <fstream>
#include <conio.h>
#define MAX 100
using namespace std;

void Input_Mat(int a[MAX][MAX], int n);
void Mat_File(char *Filename, int a[MAX][MAX], int n);

int main()
{
    int n, a[MAX][MAX];
    char filename[80];
    system("cls");
    cout<<"\nNhap so phan tu ma tran:";
    cin>>n;
    Input_Mat(a, n);
    cout<<"Nhap ten file mo de luu:";cin>>filename;
    Mat_File(filename,a,n);
    System("PAUSE");
    return 0;
}

void Input_Mat(int a[MAX][MAX], int n)
{
    int i, j;
    for ( i = 0; i < n; i++)
    {
        for (j = 0; j < n; j++)
        {
            cout<<"a["<<i<<"]["<<j<<"]="";
            cin>>a[i][j];
        }
    }
}

void Mat_File(char *filename, int a[MAX][MAX], int n)
{
    ofstream out(filename);

```

```

if (!out)
{
    cout<<"\nLoi mo file !";
    exit(1);
}
out<<n;
int i, j;
for(i = 0; i < n; i++)
{
    out<<endl;
    for(j = 0; j < n; j++)
        out<<a[i][j]<<"\t";
}
out.close();
cout<<"\nLuu file thanh cong!";
}

```

Bài 6: (Chuyển dữ liệu của tập tin cấu trúc vào Mảng cấu trúc)

Giả sử tập tin văn bản \tnhanvien.txt chứa những thông tin về nhân viên của một công ty. Mỗi dòng trong tập tin chứa các thông tin: Mã số, họ tên, Ngày tháng năm sinh, địa chỉ, Lương.

Viết hàm đọc các thông tin trong tập tin \nhanvien.txt rồi lưu trữ vào mảng 1 chiều các cấu trúc có trường tương ứng

```

#include <iostream>
#include <fstream>
#include <conio.h>
#include <string.h>
#include <iomanip>
#define MAX 20
#define THOAT 0
using namespace std;
struct date
{
    int ngay;
    int thang;
    int nam;
};
struct nhanvien
{

```

```

    int ms;
    char hoten[MAX];
    date ntn;
    char diachi[MAX];
    double luong;
};
void xuat (nhanvien ds[MAX], int n);
int read_struct(char *filename,nhanvien ds[MAX]);
//*****
int main()
{
    char filename[80];
    nhanvien ds[MAX];
    system("cls");
    cout<<"Nhập tên file mô de doc:";cin>>filename;
    int n = read_struct(filename,ds);
    xuat(ds,n);
    system("PAUSE");
    return 0;
}
int read_struct(char *filename,nhanvien ds[MAX])
{
    ifstream in(filename);
    if (!in)
    {
        cout<<"\nLỗi mô file !";
        exit(1);
    }
    int ms;
    char hoten[MAX];
    char diachi[MAX];
    double luong;
    int ngay, thang, nam;
    int i = 0;
    in>>ms; ds[i].ms = ms;
    in>>hoten; strcpy_s(ds[i].hoten,hoten);
    in>>ngay; ds[i].ntn.ngay = ngay;
    in>>thang; ds[i].ntn.thang = thang;
    in>>nam; ds[i].ntn.nam = nam;

```

```

    in>>diachi; strcpy_s(ds[i].diachi,diachi);
    in>>luong;ds[i].luong = luong;
    while (!in.eof())
    {
        i++;
        in>>ms; ds[i].ms = ms;
        in>>hoten; strcpy_s(ds[i].hoten,hoten);
        in>>ngay; ds[i].ntn.ngay = ngay;
        in>>thang; ds[i].ntn.thang = thang;
        in>>nam; ds[i].ntn.nam = nam;
        in>>diachi; strcpy_s(ds[i].diachi,diachi);
        in>>luong;ds[i].luong = luong;

    }
    in.close();
    return i;
}
void xuat (nhanvien ds[MAX], int n)
{
    cout<<setiosflags(ios:: left);
    cout<<setw(20)<<"MS"
        <<setw(20)<<"Ho Ten"
        <<setw(20)<<"NTN Sinh"
        <<setw(20)<<"Dia chi"
        <<setw(20)<<"Luong"
    cout<<endl;
    for(int i = 0; i < n; i++)
    {
        cout<<setw(20)<<ds[i].ms
            <<setw(20)<<ds[i].hoten
            <<setw(2)<<ds[i].ntn.ngay<<'/'
            <<setw(2)<<ds[i].ntn.thang<<'/'
            <<setw(16)<<ds[i].ntn.nam
            <<setw(20)<<ds[i].diachi
            <<setw(20)<<ds[i].luong
        cout<<endl;
    }
}

```

Bài 7: (Chuyển dữ liệu của Mạng cấu trúc vào tập tin)

Duyệt dữ liệu của mảng cấu trúc rồi ghi dữ liệu vào tập tin theo định dạng :

- Mỗi dòng gồm dữ liệu của các thành phần một cấu trúc.
- Trên mỗi dòng dữ liệu của các thành phần cấu trúc là tách biệt nhau (ít nhất 1 khoảng trắng)
- Khoảng trắng của các từ trong một chuỗi sẽ thay thế bằng dấu _.

Chẳng hạn, ta có một mảng các cấu trúc có kiểu NHANVIEN chứa các trường dữ liệu: Mã nhân viên, Họ tên, Năm sinh, địa chỉ, lương,

Ta chuyển dữ liệu mảng cấu trúc này vào tập tin định dạng như trên.

```
#include<iostream>
#include<fstream>
#include <iomanip>
using namespace std;

#define MAX 100
struct NHANVIEN
{
    char MaNV[10];
    char Hoten[20];
    int Namsinh;
    char Diachi[20];
    double LuongCB;
};

void MangCT_TTCT(char *f, NHANVIEN DS[MAX], int n);
int TTCT_MangCT(char *f, NHANVIEN TEST[MAX]);
void XuatMang(NHANVIEN TEST[MAX], int m);

void main()
{
    NHANVIEN DS[MAX] = {
        { "123456", "Tran_Tuan", 1960, "Da_Lat", 12500000 },
        { "103456", "Truong_Tuan_Hoc", 1961, "Sai_Gon", 12500000 },
        { "123450", "Nguyen_Van_Nam", 1970, "Nha_Trang", 20200000 }
    };

    int n = 3;
    cout << "\nDu lieu mang cau truc DS:\n";
    XuatMang(DS, n);
    system("PAUSE");
    //Chuyển mảng cấu trúc DS vào tập tin "Bai7test"
    MangCT_TTCT("Bai7test", DS, n);

    NHANVIEN TEST[MAX];
    //Chuyển lại dữ liệu tập tin "Bai7test" vào mảng cấu trúc TEST
```

```

    int m = TTCT_MangCT("Bai7test", TEST);
    cout << "\nDu lieu mang cau truc TEST:\n";
    XuatMang(TEST, m);
}
//Mang cau truc -> Tap tin cau truc
void MangCT_TTCT(char *f, NHANVIEN DS[MAX], int n)
{
    ofstream out(f);
    if (!out)
    {
        cout << "\nLoi mo file !";
        exit(1);
    }
    int i;
    for (i = 0; i < n; i++)
    {
        out << setiosflags(ios::left)
            << setw(10) << DS[i].MaNV
            << setw(20) << DS[i].Hoten
            << setw(10) << DS[i].Namsinh
            << setw(20) << DS[i].Diachi
            << setiosflags(ios::fixed) << setprecision(2)
            << setw(10) << DS[i].LuongCB << '\n';
    }
    out.close();
}
//Tap tin -> Mang cau truc
int TTCT_MangCT(char *f, NHANVIEN TEST[MAX])
{
    ifstream in(f);
    if (!in)
    {
        cout << "\nLoi mo file !";
        exit(1);
    }
    char MaNV[10];
    char Hoten[20];
    int Namsinh;
    char Diachi[20]; //Tinh
    double LuongCB;

    int i = 0;
    in >> MaNV; strcpy_s(TEST[i].MaNV, MaNV);
    in >> Hoten; strcpy_s(TEST[i].Hoten, Hoten);
    in >> Namsinh; TEST[i].Namsinh = Namsinh;

```

```

in >> Diachi; strcpy_s(TEST[i].Diachi, Diachi);
in >> LuongCB; TEST[i].LuongCB = LuongCB;

while (!in.eof())
{
    i++;
    in >> MaNV; strcpy_s(TEST[i].MaNV, MaNV);
    in >> Hoten; strcpy_s(TEST[i].Hoten, Hoten);
    in >> Namsinh; TEST[i].Namsinh = Namsinh;
    in >> Diachi; strcpy_s(TEST[i].Diachi, Diachi);
    in >> LuongCB; TEST[i].LuongCB = LuongCB;

}
i++;
in.close();
return i;
}

void XuatMang(NHANVIEN TEST[MAX], int m)
{
    int i;
    cout << setiosflags(ios::left)
        << setw(10) << "MaNV"
        << setw(20) << "Hoten"
        << setw(10) << "NS"
        << setw(20) << "Diachi"
        << setw(10) << "LuongCB";
    cout << endl;
    for (i = 0; i < m; i++)
    {
        cout << setiosflags(ios::left)
            << setw(10) << TEST[i].MaNV
            << setw(20) << TEST[i].Hoten
            << setw(10) << TEST[i].Namsinh
            << setw(20) << TEST[i].Diachi
            << setiosflags(ios::fixed) << setprecision(2) << setw(10)
            << TEST[i].LuongCB << '\n';
    }
}

```

E. LUYỆN TẬP

Chạy kiểm tra lại các bài trên.