

FICHEIROS I

Programación I

1º curso (1C)

Ficheiros

Exercicio 1

Proba o seguinte código, referido a operacións básicas sobre ficheiros.

```
#include <iostream>
#include <fstream>

#define MAXCAD 100

using namespace std;

int main() {
    char cadea [MAXCAD];
    int i,j;

    // Escritura no arquivo

    ofstream fsaida("exemplo.txt"); //fluxo de saída e apertura arquivo

    if (!fsaida) {
        cout << "Erro de apertura de arquivo" << endl;
        return 1;
    }
    else {
        fsaida << "Programacion I" << endl;
        fsaida << 15 << ' ' << hex << 15 << endl;
        fsaida.close();
    }

    // Lectura do arquivo
    ifstream fentrada ("exemplo.txt"); //fluxo de entrada e apertura arquivo

    if (!fentrada) {
        cout << "Erro de apertura de arquivo" << endl;
        return 1;
    }
    else {
        fentrada.getline(cadea, MAXCAD);
        fentrada >> i >> hex >> j;

        cout << "Contido do ficheiro: " << endl;
        cout << cadea << endl;
        cout << cadea << ' ' << i << ' ' << hex << j << endl;
        fentrada.close();
    }
    return 0;
}
```

Exercicio 2

Fai un programa que permita gardar nun ficheiro a lista das alturas dunha mostra de voluntarios. O nome do arquivo pediráselle ao usuario. Despois de abrir o ficheiro, comproba que se abriu correctamente empregando a función *fstream.fail()*, que devolverá un valor verdadeiro se houbo algún fallo na apertura.

Exercicio 3

Proba este programa de control de erros de E/S.

```
#include <iostream>
using namespace std;
void amosarIndicadores();

int main()
{
    int i;

    cout << "Estado dos indicadores para cin:"<<endl;
    amosarIndicadores ();

    cout << "Introduce un caracter (a variable e enteira): ";
    cin >> i;

    cout << "Estado dos indicadores despois do erro:"<<endl;
    amosarIndicadores ();

    cin.clear();
    cout << "Estado dos indicadores despois de borrar indicadores:"<<endl;
    amosarIndicadores ();

    return 0;
}

void amosarIndicadores()
{
    cout << "rdstate(): " << cin.rdstate() << endl;
    cout << "eof(): " << cin.eof() << endl;
    cout << "fail(): " << cin.fail() << endl;
    cout << "bad(): " << cin.bad() << endl;
    cout << "cin.good(): " << cin.good() << endl;
}
```

Exercicio 4

Fai un programa que lea os datos do arquivo xerado co exercicio anterior (deberase pedir o nome do arquivo ao usuario), e amose o listado das alturas dos voluntarios, xunto coa altura media dos mesmos no formato “A altura media dos N voluntarios é XXX”.

Exercicio 5

Proba a execución do seguinte programa, para ver os modos de acceso secuencial e directo a un arquivo.

```
#include <iostream>
#include <fstream>

using namespace std;

int main() {
    int i;

    //Así no arquivo hai bloques de 10 caracteres

    char semana[7][10] = {"Luns", "Martes", "Mercores", "Xoves", "Venres",
        "Sabado", "Domingo"};

    char cad[10];

    // Abrir arquivo para escritura e almacenar almacenar nomes dos días
    ofstream fsaida ("dias.dat", ios::out | ios::binary);
    if (fsaida.fail()) {
        cout << endl << "Erro de apertura" << endl;
        return 1;
    }
    else {
        cout << "Almacenando nomes..." << endl;

        for (i = 0; i < 7; i++) {
            fsaida.write(semana[i], 10);
        }

        fsaida.close();
    }

    // Abrir arquivo para lectura
    ifstream fentrada("dias.dat", ios::in | ios::binary);
    if (fentrada.fail()) {
        cout << endl << "Erro de apertura" << endl;
        return 1;
    }
    else {
        //Acceso secuencial
        cout << "Dias da semana:" << endl;
        fentrada.read(cad, 10);
        do {
            cout << cad << endl;
            fentrada.read(cad, 10);
        } while (!fentrada.eof());

        fentrada.clear();

        // Acceso aleatorio
        cout << "\nFin de semana:" << endl;
        for(i = 5; i <= 6; i++) {
            fentrada.seekg(10*i, ios::beg);
            fentrada.read(cad, 10);
            cout << cad << endl;
        }

        cout << "Introducir numero de 1 a 7: ";
        cin >> i;
        fentrada.seekg(10*(i-1), ios::beg);
        fentrada.read(cad, 10);
        cout << "Dia " << i << "-->" << cad << endl;
        fentrada.close();
    }

    return 0;
}
```