

Trabajar con ficheros binarios y texto

Realiza un programa que calcule la media de un conjunto de números reales positivos almacenados en un fichero. Todos los datos se guardan de nuevo en un fichero junto con la media.

```
//Programa que lee y escribe vectores desde fichero (texto) y
//calcula su media

#include <iostream.h>
#include <stdlib.h>
#include <fstream.h>
#include <string>

const int MAXIMO = 200;
typedef float Vector[MAXIMO];

//Prototipos
int F_LeerVector(ifstream &f, Vector v);
void F_EscribirVector(ofstream &f, Vector v, int tam);
void MostrarVector(Vector v, int tam);
void RestarMediaVector(Vector v, int tam, float &media);

int F_LeerVector(ifstream &f, Vector v)
{
    int i = 0;

    while( f >> v[i] && i < MAXIMO )
        i++;

    return i;
}

void F_EscribirVector(ofstream &f, Vector v, int tam)
{
    int i;

    for (i = 0; i < tam ; i++)
        f << v[i] << ' ';
    return;
}

void MostrarVector(Vector v, int tam)
{
    int i;

    cout << "El vector es:\n";
    for (i = 0; i < tam ; i++)
        cout << v[i] << ' ';
    return;
}
```

```
void RestarMediaVector(Vector v, int tam, float &media)
{
    int i;

    media = 0;
    for ( i = 0; i < tam ; i++)
        media += v[i];
    media = media / tam;

    for ( i = 0; i < tam ; i++)
        v[i] = v[i] - media;

    return;
}

int main()
{
    Vector v;
    int tam;
    ifstream f;
    ofstream fout;
    string nombrefichero;

    cout << " Este programa lee un vector de fichero\n";
    cout << "Introduce el nombre del fichero\n";
    cin >> nombrefichero;
    f.open(nombrefichero.c_str());
    if(!f)
        cout << "Error abriendo fichero\n";
    else
    {
        tam = F_LeerVector(f,v);
        f.close();
        MostrarVector(v,tam);

        CalcularMediaVector(v,media);
        cout << "Nombre de fichero para guardar resultado:\n";
        cin >> nombrefichero;
        fout.open(nombrefichero.c_str());
        if ( !fout)
            cout << "Error abriendo fichero de resultados:\n";
        else
        {
            F_EscribirVector(fout,v,tam);
            fout.close();
        }
    }

    system("PAUSE");
    return 0;
}
```