

```

/**
 * @file imagenES.cpp
 * @brief Fichero con definiciones para la E/S de imágenes
 *
 * Permite la E/S de archivos de tipo PGM,PPM
 *
 */

#include <fstream>
#include <string>

#include "imagenES.h"

using namespace std;

TipoImagen LeerTipo(ifstream& f){
    char c1,c2;
    TipoImagen res= IMG_DESCONOCIDO;

    if (f){
        c1=f.get();
        c2=f.get();
        if (f && c1=='P')
            switch (c2) {
                case '5': res= IMG_PGM; break;
                case '6': res= IMG_PPM; break;
                default: res= IMG_DESCONOCIDO;
            }
    }
    return res;
}

// _____

TipoImagen LeerTipoImagen(const char *nombre){
    ifstream f(nombre);
    return LeerTipo(f);
}

// _____

char SaltarSeparadores (ifstream& f){
    char c;
    do{
        c= f.get();
    } while (isspace(c));
    f.putback(c);
    return c;
}

// _____

bool LeerCabecera (ifstream& f, int& fils, int& cols){
    int maxvalor;
    string linea;
    while (SaltarSeparadores(f)=='#')
        getline(f,linea);
    f >> cols >> fils >> maxvalor;

    if (/*str &&*/ f && fils>0 && fils<5000 && cols>0 && cols<5000){
        f.get(); // Saltamos separador
        return true;
    }
    else
        return false;
}

// _____

```

```

unsigned char *LeerImagenPPM (const char *nombre, int& fils, int& cols){
    unsigned char *res=0;
    fils=0;
    cols=0;
    ifstream f(nombre);

    if (LeerTipo(f)==IMG_PPM){
        if (LeerCabecera (f, fils, cols)){
            res= new unsigned char[fils*cols*3];
            f.read(reinterpret_cast<char *>(res),fils*cols*3);
            if (!f){
                delete[] res;
                res= 0;
            }
        }
    }
    return res;
}

```

---

```
//
```

```

unsigned char *LeerImagenPGM (const char *nombre, int& fils, int& cols){
    unsigned char *res=0;
    fils=0;
    cols=0;
    ifstream f(nombre);

    if (LeerTipo(f)==IMG_PGM){
        if (LeerCabecera (f, fils, cols)){
            res= new unsigned char[fils*cols];
            f.read(reinterpret_cast<char *>(res),fils*cols);
            if (!f){
                delete[] res;
                res= 0;
            }
        }
    }
    return res;
}

```

---

```
//
```

```

bool EscribirImagenPPM (const char *nombre, const unsigned char *datos,
                        const int fils, const int cols){
    ofstream f(nombre);
    bool res= true;

    if (f){
        f << "P6" << endl;
        f << cols << ' ' << fils << endl;
        f << 255 << endl;
        f.write(reinterpret_cast<const char *>(datos),fils*cols*3);
        if (!f)
            res=false;
    }
    return res;
}

```

---

```
//
```

```

bool EscribirImagenPGM (const char *nombre, const unsigned char *datos,
                        const int fils, const int cols){
    ofstream f(nombre);
    bool res= true;

    if (f){
        f << "P5" << endl;
        f << cols << ' ' << fils << endl;
        f << 255 << endl;
        f.write(reinterpret_cast<const char *>(datos),fils*cols);
        if (!f)
            res=false;
    }
}

```

```
    }  
    return res;  
}
```

```
/* Fin Fichero: imagenES.cpp */
```