BMP文件结构C实现和C++实现的异同

```c
//C版
#include "stdio.h"
#include "math.h"
#include "windows.h"

#ifndef BMPFILEC
#define BMPFILEC

    extern BYTE *Imagedata;
    extern int imagew,imageh;
    extern int iYRGBnum;
    extern RGBQUAD palette[256];

    BYTE *pDataAt(int height,int RGB = 0);
    BOOL AllocateMem();

    BOOL LoadBMPFILE(char *fname);
    BOOL SaveBMPFILE(char *fname);

    void Init();
    void Release();

#endif
```

```c
//C++版
#include "stdio.h"
#include "math.h"
#include "windows.h"

#ifndef HXLBMPFILEH
#define HXLBMPFILEH
class HXLBMPFILE
{
    BYTE *Imagedata;

public:
    int imagew,imageh;
    int iRGBnum;

    RGBQUAD palette[256];

    BYTE *pDataAt(int height,int RGB = 0);
    BOOL AllocateMem();

    BOOL LoadBMPFILE(char *fname);
    BOOL SaveBMPFILE(char *fname);

    HXLBMPFILE();
    ~HXLBMPFILE();
};
#endif
```

```c
//C版
// bmpfileC.cpp : Defines the entry point for the console application.
//
#include "stdafx.h"
#include "bmpfileC.h"
BYTE *Imagedata;
int imagew,imageh;
int iYRGBnum;
RGBQUAD palette[256];
void Init()
{
    Imagedata=NULL;

    for (int i = 0; i< 256;i++)
    {
        palette [i].rgbBlue =
        palette [i].rgbGreen =
        palette [i].rgbRed = i;
        palette [i].rgbReserved = 0;
    }

    iYRGBnum = 0;
    imagew = imageh = 0;
}

void Release() { ... }

BYTE *pDataAt(int height, int RGB) { ... }

BOOL AllocateMem() { ... }

BOOL LoadBMPFILE (char *cFilename) { ... }

BOOL SaveBMPFILE (char *cFilename)
{
    if (!Imagedata) return FALSE;

    FILE *f = NULL;
```

```c
//C++版
// bmpfile.cpp : Defines the entry point for the console application.
//
#include "stdafx.h"
#include "bmpfile.h"

HXLBMPFILE::HXLBMPFILE()
{
    Imagedata=NULL;

    for (int i = 0; i< 256;i++)
    {
        palette [i].rgbBlue =
        palette [i].rgbGreen =
        palette [i].rgbRed = i;
        palette [i].rgbReserved = 0;
    }

    iRGBnum = 0;
    imagew = imageh = 0;
}

HXLBMPFILE::~HXLBMPFILE() { ... }

BYTE *HXLBMPFILE::pDataAt(int height, int RGB) { ... }

BOOL HXLBMPFILE::AllocateMem() { ... }

BOOL HXLBMPFILE::LoadBMPFILE (char *cFilename) { ... }

BOOL HXLBMPFILE::SaveBMPFILE (char *cFilename)
{
    if (!Imagedata) return FALSE;

    FILE *f = NULL;
```

```
//彩色变灰度（C版）
int i,j;
Init();  //Init里面实现了给灰度图像设定调色板
if( !LoadBMPFILE("firstpage-color.bmp") ) return 1;
int iYRGBnumN = 1;
BYTE* imagedataN = NULL;
imagedataN = new BYTE[imagew*imageh];
if( !imagedataN ) return 1;
for (i=0;i<imageh;i++)
{
    for (j=0;j<imagew;j++)
    {
        //彩色变灰度
        imagedataN[i*imagew+j] =BYTE( 0.299*pDataAt(i,0)[j]+0.587*pDataAt(i,1)[j]+0.114*pDataAt(i,2)[j]+0.5);
    }
}
delete[] Imagedata;
Imagedata = NULL;
Imagedata = imagedataN;
iYRGBnum = iYRGBnumN;
SaveBMPFILE ("firstpage-gray.bmp");
Release();
```

```
//彩色变灰度（C++版）
HXLBMPFILE bf;
int i,j;
if (!bf.LoadBMPFILE ("hometown.bmp")) return 1;
HXLBMPFILE bmpfile;
bmpfile.imagew = bf.imagew;
bmpfile.imageh = bf.imageh;
bmpfile.iRGBnum = 1;
if (!bmpfile.AllocateMem()) return 1;
for (i=0;i<bf.imageh;i++)
    for (j=0;j<bf.imagew;j++)
    {
        bmpfile.pDataAt(i,0)[j] =BYTE( 0.299*bf.pDataAt (i,0)[j]+0.587*bf.pDataAt (i,1)[j]+0.114*bf.pDataAt (i,2)[j]+0.5);
    }
bmpfile.SaveBMPFILE ("hometown-gray2.bmp");
```