**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP KỸ THUẬT LẬP TRÌNH**

**ĐỌC GHI FILE BMP**

HỌ VÀ TÊN: BÙI HỒNG DƯƠNG

MSSV: 20120273

Trong bài làm của em gồm có các cấu trúc như sau:

+ struct header: lưu phần header của ảnh

+ struct DIB: lưu phần DIB của ảnh

+ struct mau: lưu các chỉ số màu A,B(xanh lục),G(xanh lam) ,R(đỏ) của 1 pixel

+ struct AnhBMP: để đơn giản bớt tham số truyền vào các hàm, em bỏ tất cả các thông tin của ảnh vào cùng 1 struct. Bao gồm header, DIB, phần dữ liệu điểm ảnh (kiểu unsigned char) và phần dư( khoảng nằm giữa 40 byte của DIB và phần dữ liệu điểm ảnh)

Tiếp đến là các hàm:

+ void docanh(char\* file, AnhBMP& anh): để đọc được 1 ảnh thì ta cần đường dẫn file input( chính là char\* file).Tiếp theo đó là 1 struct AnhBMP để lưu các thông tin của ảnh.

+void ghianh(char\* file, AnhBMP anh): để lưu ảnh BMP xuống 1 file, ta cần truyền vào đường dẫn file đó và thông tin ảnh BMP

+void chuyensanganh8bpp(char\* fileoutput,AnhBMP anh): hàm này để chuyển 1 ảnh từ 24/32 bpp xuống 8 bpp.Ta cần truyền vào đường dẫn lưu file và thông tin của ảnh BMP 24/32 bpp (AnhBMP anh).

+void chuyendatasangpixel(AnhBMP anh,mau\*& pixel): vì cái dữ liệu điểm ảnh được lưu trong 1 mảng unsigned char data nên rất khó để truy xuất vào đúng điểm dữ liệu nào lưu màu B,G,R,A. Vì thế chúng ta sẽ chuyển sang 1 mảng pixel: trong đó 1 phần tử pixel sẽ lưu dữ liệu điểm ảnh của 3 màu (B,G,R) nếu là 24 bpp và 4 màu( A,B,G,R) nếu là 32 bpp.

+void chuyentupixelsangdata(AnhBMP anh,mau\* pixel): sau khi đã xử lý xong với các dữ liệu điểm ảnh dưới dạng pixel, ta cần chuyển nó về lại dạng mảng unsigned char data để lưu ảnh.

+void resize(AnhBMP anh,int s,char\* fileoutput): hàm này là để thu nhỏ ảnh theo 1 tỷ lệ S cho trước.Chúng ta cần truyền vào thông tin ảnh( AnhBMP anh), tỷ lệ S(int s), đường dẫn lưu file(char\* fileoutput).