워터마크 삽입 프로그램

Iwannabegosu조 박목월, 김태윤

```
struct png{
    png_uint_32 width;
    png_uint_32 height;
    png_bytepp buf;
    png_byte color_type;
    png_byte color_depth;
};
```

```
png* ReadPNG(const char *filename)
    png_byte signature[8] = {0, };
    png *pinfo = NULL;
 if(pinfo->color_type == PNG_COLOR_TYPE_RGB){
    png_bytepp tmp = (png_bytepp)malloc(sizeof(png_bytep) * pinfo->height);
    for(unsigned int y = 0; y < pinfo->height; y++){
        tmp[y] = (png_bytep)malloc((pinfo->width) * PNG_COLOR_TYPE_RGBA);
        for(unsigned int x = 0; x < pinfo->width; x++){
            tmp[y][x * 4] = pinfo->buf[y][x * 3];
            tmp[y][x * 4 + 1] = pinfo->buf[y][x * 3 + 1];
            tmp[y][x * 4 + 2] = pinfo->buf[y][x * 3 + 2];
            tmp[y][x * 4 + 3] = 255;
        }
    }
    if(pinfo->buf){
        for(unsigned int i = 0; i < pinfo->height; i++)
            if(pinfo->buf[i]) free(pinfo->buf[i]);
        free(pinfo->buf);
    }
    pinfo->buf = tmp;
    pinfo->color_type = PNG_COLOR_TYPE_RGBA;
 }
 if(type){
    for(unsigned int y = 0; y < pinfo->height; y++){
        for(unsigned int x = 0; x < pinfo->width; x++){
            if(pinfo->buf[y][x * 4] == 255 && pinfo->buf[y][x * 4 + 1] == 255 && pinfo->buf[y][x * 4 + 2] == 255)
                pinfo->buf[y][x * 4 + 3] = 0;
        }
    }
 }
        fclose(fp);
        return NULL;
    }
```

```
void AlphaBlending(png* target, png* logo, unsigned int sy, unsigned int sx, double alpha){
    for(unsigned int y = 0; y < logo->height; y++){
        for(unsigned int x = 0; x < logo->width; x++){
            if(logo->buf[y][x * 4 + 3]){
                double r = (double) target->buf[y + sy][(sx + x) * 4] * (1 - alpha) + (double) logo->buf[y][x * 4] * alpha;
                double g = (double) target->buf[y + sy][(sx + x) * 4 + 1] * (1 - alpha) + (double) logo->buf[y][x * 4 + 1] * alpha;
                double b = (double) target->buf[y + sy][(sx + x) * 4 + 2] * (1 - alpha) + (double) logo->buf[y][x * 4 + 2] * alpha;
                double a = (double) target->buf[y + sy][(sx + x) * 4 + 3] * (1 - alpha) + (double) logo->buf[y][x * 4 + 3] * alpha;
                target->buf[y + sy][(sx + x) * 4] = (int) r;
                target->buf[y + sy][(sx + x) * 4 + 1] = (int) g;
                target->buf[y + sy][(sx + x) * 4 + 2] = (int) b;
                target->buf[y + sy][(sx + x) * 4 + 3] = (int) a;
            }
            if(x + 1 + sx == target->width * 4) break;
        }
        if(y + sy + 1 == target->height) break;
    }
    return;
}
```

```
double alpha = alp / (double) 100;
switch(type){
    case 1:
        AlphaBlending(target, logo, 0, 0, alpha);
        break;
    case 2:
        AlphaBlending(target, logo, 0, target->width - logo->width - 1, alpha);
        break;
    case 3:
        AlphaBlending(target, logo, target->height - logo->height - 1, 0, alpha);
        break;
    case 4:
        AlphaBlending(target, logo, target->height - logo->height - 1, target->width - logo->width - 1, alpha);
        break;
    case 5:
        for(unsigned int sy = 0; sy < target->height; sy += logo->height){
            for(unsigned int sx = 0; sx < target->width; sx += logo->width){
                AlphaBlending(target, logo, sy, sx, alpha);
            }
        }
        break;
}
```

```c
int main(int argc, char* argv[]){
    char targetname[256], logoname[256];
    printf("Please provide the file name of the image to which the watermark will be applied:");
    scanf("%s", targetname);
    printf("Please provide the file name of the watermark image:");
    scanf("%s", logoname);
    png* target = ReadPNG(targetname);
    png* logo = ReadPNG(logoname);
    if(logo->height > target->height || logo->width > target->width){
        printf("Target Picture is Too small");
        return 0;
    }
    int type, alpha;
    printf("Type 1: insert watermark in the top left corner\n");
    printf("Type 2: Insert watermark in the top right corner\n");
    printf("Type 3: Insert watermark in the bottom left corner\n");
    printf("Type 4: Insert watermark in the bottom right corner\n");
    printf("Type 5: Insert watermark across the entire image\n");
    printf("Please choose the desired watermark image insertion method.\n");
    scanf("%d", &type);
    printf("Please specify the opacity of the watermark in percentage:");
    scanf("%d", &alpha);
    InsertWatermark(target, logo, type, alpha);
    FreePNG(target);
    FreePNG(logo);
    return 0;
}
```