

CS1103
Programación Orientada a Objetos II
Practica Calificado 3b
2019 - 2

Profesor: Rubén Rivas

Alumno:

Los archivos de imagen tienen diferentes formatos BMP, JPG o PNG, algunos con capacidad de compresión otros sin esta capacidad, pero todos ellos tienen en común que almacenan en una sección la información sin codificar o codificada de cada uno de los píxeles.

Cada píxel es representado usualmente por 3 o 4 bytes dependiendo del formato, si es que soporta transparencia o no, los 3 o 4 bytes representan la combinación de 3 colores y un nivel de transparencia (RGBA, Red, Green, Blue y Alpha) siendo Alpha el nivel de transparencia que es opcional en aquellos casos que no se soporta transparencia.

Se solicita crear un programa que permita leer archivos PNG y que realice las siguientes operaciones con el archivo.

- Rotar el archivo
- Filtrar colores

Es obligatorio que estas operaciones se realicen concurrentemente, lo que se sugiere que la imagen se subdivide en partes 4 o 16 partes y que cada parte sea procesada en diferente thread.

Input

La información será recibida en un archivo (*in.png*)

Output

Deberán generarse 4 archivos con formato PNG (*rotated.png*, *red_filter.png*, *red_filter.png*, *blue_filtered.png*):

- **rotated.png**, contendrá la información rotada a 45° (https://en.wikipedia.org/wiki/Rotation_matrix).
- **red_filter.png**, mantendrá el componente **Red** y **Alpha** de los píxeles y el resto de los componentes será modificado a 0.
- **green_filter.png**, mantendrá el componente **Green** y **Alpha** de los píxeles y el resto de los componentes será modificado a 0.

- **blue_filter.png**, mantendrá el componente **Blue** y **Alpha** de los pixeles y el resto de los componentes será modificado a 0.

Nota

Para facilitar la lectura del archivo se ha provisto 2 funciones de ejemplo y se ha incluido la librería lodePNG (<https://lodev.org/lodepng/>).

El contenido de la imagen es representada por un vector (`std::vector<unsigned char>`) donde cada píxel es representado por un set de 4 **unsigned char** en este orden (**RGBA**) ver código ejemplo. Al momento de rotar debe tomarse en cuenta que esos 4 **unsigned char** deben moverse simultáneamente.

Requerimiento	Puntos
Lectura correcta del archivo	2
set de pruebas adecuado	2
Obtención correcta de los resultados, organizado adecuada y racionalmente en archivos .h y .cpp , con mínima cantidad de sentencias no más de 25 instrucciones por función o método.	5
Generación de archivos correctamente	2
Implementación concurrente del rotado de imagen	3
Implementación concurrente del filtrado de imagen	3
Uso adecuado de la librería estándar, clases y template si se requiera.	3
Total:	20

Barranco, 1ero de diciembre del 2019