

Proyecto de Estructura de Datos:

Componente 1 – Plan de Pruebas

Esteban Navas

Juan Bello

Santiago Hernández

Pontifica Universidad Javeriana

Resumen

En este documento se desarrolla el plan de pruebas para comprobar que el componente 1 utilizado en el proyecto de estructura de datos funcione como se supone debería funcionar, permitiendo la corrección de posibles errores, se llevara a cabo la planeación de resultados esperados y resultados obtenidos de la función `proyeccion2D`.

Compilación y Ejecución

Pasos realizados:

1. compilación de los archivos imagen.h proyección.h proyección.cpp, imagen.cpp, main.cpp con información de depuración.
2. `G+ -std-c++11 -o prog *.cpp`
3. Ejecución del programa
4. `./prog`
5. Carga de volumen con propósito de procesar en memoria con comandos de consola
6. Comando: `cargar_volumen + nombre_imagen_base + numero de imagenes`
7. Procesamiento del volumen en memoria con comandos de consola
8. Comando: `Proyeccion2D`
9. Resultado:

```
vboxuser@Ubuntu:~/Estructuras_D/proyecto$ g++ -o prog *.cpp
vboxuser@Ubuntu:~/Estructuras_D/proyecto$ ./prog
$ cargar_volumen IM-126-0002-epiT2 36

Se cargó IM-126-0002-epiT2 con 36 imágenes exitosamente

se cargo IM-126-0002-epiT2 con 36 imagenes exitosamente

$ info_volumen

IM-126-0002-epiT2
ancho: 256
alto: 256
tamaño: 36
$ proyeccion2D x minimo result.pmg

La proyección 2D ha sido generada y almacenada en result.pmg

$ proyeccion2D x maximo result.pmg

La proyección 2D ha sido generada y almacenada en result.pmg

$
```

Imagen 1. Salida del programa

Plan de Pruebas

Función Proyección2D

- Descripción:
 - La función Proyeccion2D recibe como parámetros la dirección en la que se desea generar la proyección (X , Y o Z), el parámetro con el que se desea generar(Mínimo, Máximo, Promedio) y el nombre del archivo en el que se guardara la proyección, Proyeccion2D por si sola se encarga de generar la proyección pero no se encarga de guardarla en memoria, de eso se encargara la función guardarPMG que recibe de Proyeccion2D el nombre de la imagen resultante.
- Resultados esperados:

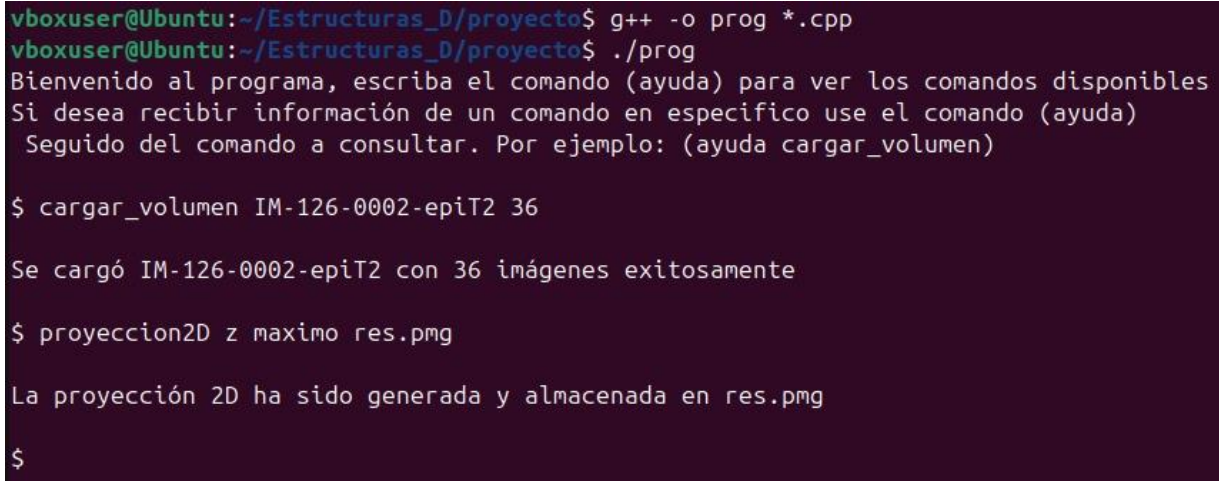
Descripción de entrada	Valores esperados	Resultados obtenidos	Resultado de Caso
1. Volumen cargado en memoria.	Vol, Comando [1], comando [2]	La Proyección 2D ha sido generada y almacenada en nombre_imagen.pmg	La Proyección 2D ha sido generada y almacenada en nombre_imagen.pmg
2: Volumen no cargada en memoria	Vol, comando[1], comando[2]	El volumen aún no ha sido cargado en memoria	La Proyección 2D ha sido generada y almacenada en nombre_imagen.pmg

Tabla 1. Resultados plan de prueba

Análisis de Errores y Corrección

Con el código implementado originalmente no se detectaron errores por lo que no fue necesario implementar una corrección al código.

Prueba 1:



```
vboxuser@Ubuntu:~/Estructuras_D/proyecto$ g++ -o prog *.cpp
vboxuser@Ubuntu:~/Estructuras_D/proyecto$ ./prog
Bienvenido al programa, escriba el comando (ayuda) para ver los comandos disponibles
Si desea recibir información de un comando en especifico use el comando (ayuda)
Seguido del comando a consultar. Por ejemplo: (ayuda cargar_volumen)

$ cargar_volumen IM-126-0002-epiT2 36

Se cargó IM-126-0002-epiT2 con 36 imágenes exitosamente

$ proyeccion2D z maximo res.png

La proyección 2D ha sido generada y almacenada en res.png

$
```

Imagen 2. Resultado prueba

Conclusiones de la Prueba

- Se logro compilar y ejecutar proyección.cpp, imagen.cpp, main.cpp sin errores.
- Se logro cargar volúmenes y obtener su información en el sistema.
- Se logro generar la proyección2d del volumen en memoria.
- Se logro obtener la información del volumen en memoria.
- El código final cumple con los requisitos necesarios para el funcionamiento del componente 1 del Proyecto de Estructuras.