Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Preprocesado**

Redirecciono el contenido a un archivo main.i (es el código del preprocesado). Todo el contenido de las librerías #include se ponen ahí.

g++ -E .\main.cpp > main.i



**Compilado**

Pasa el código a código assembler

g++ -S .\main.i



**Ensamblado**

Creamos el archivo que está en código binario

g++ -c .\main.s



**Enlazado**

Crea el archivo .exe

g++ .\main.o -o main

**Cabecera**

Es un archivo de extensión .h o .hpp donde se declaran las funciones que se van a usar.

Texto

Descripción generada automáticamente

Solo se incluye la cabecera

**Definición**

Es un archivo .cpp donde se define el comportamiento de cada función declarada en la cabecera.

Pantalla de computadora con letras

Descripción generada automáticamente con confianza media

#include “cabecera.h”

**Compilación**

*Solo los archivos .cpp se compilan*

g++ -c .\operaciones.cpp

g++ -c .\main.cpp

**Enlazado**

*Todos los archivos compilados (.o) que tengamos los enlazamos al main (.exe)*

g++ .\operaciones.o .\main.o -o main

**Ejecución**

.\main.exe

**Makefile**

Es un archivo para automatizar el proceso de compilación y ejecución de nuestro proyecto.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente



Creamos un archivo llamado “Makefile” (por convención con “m” mayúscula) en el proyecto.

**all: build** define a **build** como valor por defecto

**clean** es la otra función que definimos para poder eliminar los archivos que se generan (.o y .exe)

Captura de pantalla de computadora

Descripción generada automáticamente

Llamamos al archivo escribiendo “make” en el terminal. Ejecuta el **build**.

Texto

Descripción generada automáticamente

Para llamar a la otra funcionalidad, usamos **“make clean”**

