



APP1 Lenguajes y Paradigmas de Programación

Estudiantes:

Vicente Vásquez

Pascal Jorquera

Profesor:

Justo Vargas

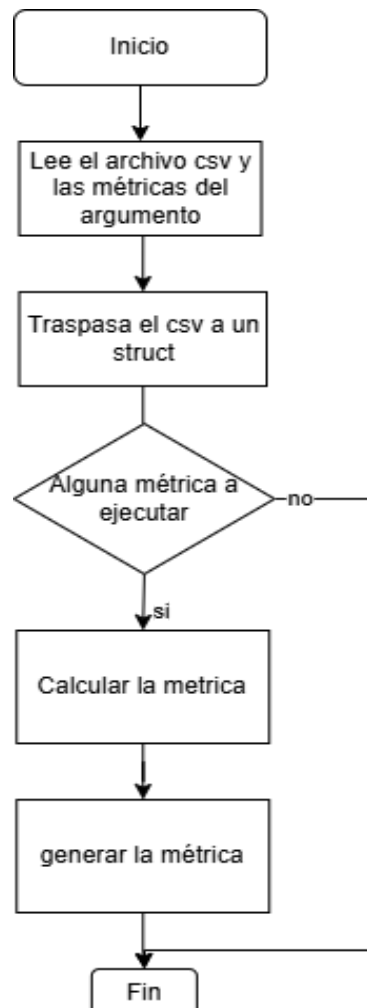
Fecha: 5 de Abril de 2025

Organización del código:

El código está dividido en múltiples archivos para mantener un orden claro y cumplir con el paradigma procedural:

- main.c: Archivo principal del programa
- orders.h: Define un struct para almacenar dinámicamente los datos del csv
- orders.c: Abre y lee el archivo csv y carga los datos en la memoria
- metrics.h: Define todas las métricas, con el formato de puntero char* *métrica*(int* size, order* orders):
 - ◆ Pizza más/menos vendida (pms, pls)
 - ◆ Fecha con más/menos ventas (dms, dls, dmsp, dlsp)
 - ◆ Promedios (apo, apd)
 - ◆ Ingrediente más vendido (ims)
 - ◆ Pizzas por categoría (hp)
- metrics.c: Contiene la lógica de ejecución de cada métrica solicitada por el usuario
- Makefile: Automatización de compilación, permite compilar con “make start”
- ReadMe: Información del grupo

Diagrama de flujo del programa:



Reflexiones:

Como grupo, uno de los principales desafíos a la hora de crear la aplicación, fue poder compilar el programa. Durante el proyecto tuvimos repetidos errores de compilación, ya sea de la estructura del código, problemas de declaración de punteros, problemas en reconocer las métricas y al momento de compilar el programa nos entregaba valores erróneos o simplemente no se ejecutaba. Tras varios intentos fallidos, la solución fue instalar Ubuntu que es un subsistema de Linux para Windows, lo que nos permitió un entorno más compatible con C y así poder compilar de forma correcta la aplicación.

Explicación IA:

En este trabajo se utilizó Chat GPT y GitHub Copilot, para obtener sugerencias en la estructura modular del programa (división de archivos .c y .h), también se usó para corrección y explicación de errores de compilación. Para validar esta información se probó en el programa y nos aseguramos que las sugerencias de la IA estuvieran de la mano con la lógica del problema.