

Practica 1. 2021/1 Laboratorio de Sistemas Operativos

March 11, 2021

1 Planteamiento del problema

El restaurante **Tapioquería Tigarto** ofrece una amplia variedad de platos a sus distinguidos comensales. El propietario del restaurante, cariñosamente conocido por sus clientes como el **Profe Patterson**, se ha sabido craneanar lo que espera sea una exitosa campaña de mercadeo a propósito del jjiuemadre COVID-19 (es que desde que comenzó la pandemia le hacen pedidos a domicilio como un berraco). Dado que su fiel clientela confía ciegamente en la calidad de sus platos, el Profe Patterson ha comenzado a ofrecer a sus clientes un servicio en el que el cliente solo debe indicar la cantidad de platos que desea en su domicilio y el personal de Tapioquería Tigarto se encargará de asignar a cada pedido los platos que garantizan la mayor cantidad de ingredientes posibles. El slogan de la campaña versa: *Tapioquería Tigarto. En la variedad está el placer.*

El Profe Patterson (que además de gerenciar el restaurante da clases en la U) le encarga a sus estudiantes de programación la codificación de un programa escrito en lenguaje C que se encargue de determinar la asignación de platos a cada pedido de manera que se logre el propósito de ofrecer la mayor cantidad posible de ingredientes a sus clientes. Hay ciertas limitaciones que los estudiantes deben tener en cuenta para diseñar y codificar su programa:

- Los clientes solo pueden realizar pedidos de dos, tres o cuatro platos (si usted quiere pedir 5 platos, pues haga un pedido de 3 y otro de 2 y listo; y si solo necesita un plato, pues pida de una vez el almuerzo y la comida)
- La cantidad de platos pedidos corresponde a *sendos* platos ya preparados y disponibles en el restaurante, cada uno con una cantidad de ingredientes diferentes (la primera vez que escuché la palabra sendos fue en la clase de teoría electromagnética con el Mazo Gaviria, muy útil pa esta frase, jejeje)

Cada vez que el Profesor Patterson ejecuta el programa le pasa como argumento un archivo con el siguiente formato (a continuación, un archivo de ejemplo):

```
9 1 1 1
5 champi cebolla pimenton jamon ajo
5 huevo cilantro aji ajo jamon
4 cebolla queso garbanzo aji
5 champi cebolla pimenton jamon ajo
5 huevo cilantro aji ajo jamon
4 cebolla queso garbanzo aji
4 cebolla queso garbanzo aji
4 cebolla queso garbanzo aji
4 cebolla queso garbanzo aji
```

La primera línea consta de 4 enteros separados por un espacio en blanco. El primer número indica la cantidad de platos pedidos, el segundo número indica la cantidad de pedidos de dos platos que se deben despachar, el tercer número indica la cantidad de pedidos de tres platos que se deben despachar y el cuarto número indica la cantidad de pedidos de cuatro platos que se deben despachar. Es decir, dos veces el segundo número sumando con tres veces el tercer número y sumado con cuatro veces el cuarto número debe ser igual al primer número (en el ejemplo $2*1 + 3*1 + 4*1 = 9$). Las líneas siguientes del archivo describen cada plato; es decir que si excluimos la primera línea, el número de líneas del archivo debe ser igual al primer número de la primera línea (en el ejemplo, el archivo tiene diez líneas, excluyendo la primera son nueve,

justamente el primer número de la primera línea). Cada línea empieza con un número que indica la cantidad de ingredientes del correspondiente plato seguido de la lista de ingredientes. Cada ingrediente puede tener hasta 20 caracteres alfanuméricos, todos en minúscula y sin caracteres especiales (es decir, sin tildes ni ñes, etc...). Solo se permite el carácter _ para separar las palabras de ingredientes que sean más de una palabra.

Como es bien sabido por ustedes, cada línea del archivo termina con un caracter de nueva línea. Obsérvese además que a pesar de que los últimos 4 platos del ejemplo son iguales todos estos se incluyen. Es decir, si el Profe Patterson ordenó preparar 20 sancochos, pues habrá 20 líneas correspondientes a los 20 sancochos, cada una con los mismos ingredientes. Puede pasar que los ingredientes estén en desorden pero si son los mismos entonces corresponden al mismo plato; es decir, los siguientes dos platos son iguales:

5 cebolla champi pimenton jamon ajo
5 champi cebolla jamon pimenton ajo

2 Modelo matemático

Para facilitar la tarea a sus estudiantes, el Profe Patterson ha decidido entregarles el modelo matemático correspondiente al problema indicado. A continuación lo describimos:

- Sea PP la cantidad de platos pedidos (es decir el primer entero de la primera línea del archivo de entrada)
- Sea I la cantidad de ingredientes diferentes en la totalidad de los platos (los estudiantes deben calcularlo según el archivo de entrada)
- P_2 la cantidad de pedidos de dos platos (el segundo numero de la primera línea del archivo)
- P_3 la cantidad de pedidos de tres platos (el tercer numero de la primera línea del archivo)
- P_4 la cantidad de pedidos de cuatro platos (el cuarto numero de la primera línea del archivo)
- $NTPP$ el número total de platos pedidos, es decir $NTPP = 2 * P_2 + 3 * P_3 + 4 * P_4$
- La matriz $P[PP][I]$ es la matriz de platos de forma tal que $P_{i,j} = 1$ si el ingrediente j está presente en plato i y $P_{i,j} = 0$ en otro caso.
- $AP[NTPP]$ el vector que indicará la asignación de platos para cada pedido (el índice indica un plato correspondiente a un pedido y el valor en la posición indicada por ese índice denota el plato asignado)
- Y finalmente sea $OP_3 = 2 * P_2$ el offset a partir del cual empiezan en el vector AP los platos para los pedidos de tres platos y $OP_4 = 2 * P_2 + 3 * P_3$ el offset a partir del cual empiezan en el vector AP los platos para los pedidos de cuatro platos

El programa debe encontrar el vector AP que maximiza la siguiente función:

$$\begin{aligned}
 &MAXIMIZE(\\
 &\quad \sum_{i=0}^{P_2-1} \sum_{j=0}^{I-1} OR \{P(AP[2 * i], j); P(AP[2 * i + 1], j)\} + \\
 &\quad \sum_{i=0}^{P_3-1} \sum_{j=0}^{I-1} OR \{P(AP[OP_3 + 3 * i], j); P(AP[OP_3 + 3 * i + 1], j); P(AP[OP_3 + 3 * i + 2], j)\} + \\
 &\quad \sum_{i=0}^{P_4-1} \sum_{j=0}^{I-1} OR \{P(AP[OP_4 + 4 * i], j); P(AP[OP_4 + 4 * i + 1], j); P(AP[OP_4 + 4 * i + 2], j); P(AP[OP_4 + 4 * i + 3], j)\} \\
 &\quad)
 \end{aligned}$$

Obsérvese que considerando las definiciones indicadas al inicio de esta sección, la primera doble sumatoria corresponde a la cuenta de los ingredientes diferentes que fueron asignados a los pedidos de dos platos, la segunda doble sumatoria a la cuenta de los ingredientes diferentes que fueron asignados a los pedidos de tres platos y finalmente la tercera doble sumatoria a la cuenta de los ingredientes diferentes que fueron asignados a los pedidos de cuatro platos.

3 Entregables de la práctica

Cada equipo debe entregar un pequeño informe de no más de dos páginas (pillen muchachos que es pequeño el informe así que tómense el trabajo de redactarlo bien, puede ser en lenguaje coloquial como esta guía, pero bien escrita por favor ¹) describiendo la estrategia que adoptó para resolver el problema, los principales retos que se encontraron en el desarrollo de la práctica y un pequeño análisis de las principales diferencias que encontró entre el lenguaje C y los lenguajes de alto nivel que ustedes utilizan habitualmente; eso último obviamente referido al desarrollo de la práctica. Además debe entregar un archivo (o varios) con extensión .c o .h correspondiente al código de su programa. El programa debe recibir como parámetro la ruta del archivo con la descripción de los platos y los pedidos (un archivo con el mismo formato al descrito arriba, aunque puede tener un contenido diferente) y debe presentar el resultado en un archivo nombrado como "output.txt" con el siguiente formato de salida:

Esta es la matriz de los platos:

```
1 0 0 1 0 0 0 0 0
1 0 1 1 0 1 1 1 1
1 0 0 1 0 0 0 0 0
1 1 0 1 1 0 0 0 0
1 1 0 1 1 0 0 0 0
0 1 0 0 1 0 0 0 0
0 1 0 0 1 0 0 0 0
0 1 1 0 1 1 1 1 1
0 0 1 0 0 1 1 1 1
0 0 1 0 0 1 1 1 1
```

Este es el vector de solución: 0 2 1 3 5 4 6 7 8

La cantidad de ingredientes diferentes totales es: 26

El pedido 0 contiene: champi ,cebolla ,pimenton ,jamón ,ajo ,ají ,queso ,garbanzo ,

El pedido 1 contiene: champi ,cebolla ,pimenton ,jamón ,ajo ,huevo ,cilantro ,ají ,queso ,garbanzo ,

El pedido 2 contiene: cebolla ,jamón ,ajo ,huevo ,cilantro ,ají ,queso ,garbanzo ,

Nota: Recuerde blindar su programa a los errores de entrada; por ejemplo cuando el usuario no envía una ruta como argumento de entrada o cuando el archivo de entrada no cumple con las restricciones indicadas en la explicación del correspondiente formato.

¹Si se pillan algún error en la redacción por favor háganmelo saber que la idea es irse puliendo uno también, jejeje.