

Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

IIC1103 – Introducción a la Programación
2 - 2017

Interrogación 1

Instrucciones generales:

- **Para resolver la interrogación solo necesitas hasta la sección 8 del Recordatorio de contenidos adjunto.**
- La interrogación consta de 3 preguntas con igual ponderación.
- Solo se reciben consultas sobre el enunciado en los primeros 30 minutos.
- La última página es un recordatorio; no está permitido utilizar material de apoyo adicional.
- Todas las preguntas están pensadas para ser resueltas en forma sencilla con lo visto en clases. No es necesario utilizar otros mecanismos más avanzados.
- Si el enunciado de la pregunta no lo indica explícitamente, no es necesario validar los ingresos del usuario. Puedes suponer que no hay errores por ingreso de datos.
- Responde cada pregunta en una hoja separada, marcando tu Número de alumno y el número de pregunta en todas las hojas.
- Puedes utilizar cualquier cosa que te hayan pedido programar en un ítem anterior dentro de las preguntas (incluso si no respondiste ese ítem).
- Tienen dos horas para resolver la prueba desde que se entregan los enunciados.

Pregunta 1 (1/3)

¡Algo terrible ha ocurrido!. Tu aplicación preferida de mensajería, Introgram, enloqueció y está mezclando los mensajes de tus contactos.

Cada vez que dos usuarios distintos te envían un mensaje, Introgram los mezcla y te muestra solamente un mensaje. En el mensaje mezclado, los primeros 2 caracteres corresponden al primer mensaje, los siguientes 2 caracteres corresponden al segundo mensaje, los siguientes 3 caracteres corresponden al primer mensaje, los siguientes 3 caracteres corresponden al segundo mensaje, los siguientes 4 caracteres corresponden al primer mensaje, y así sucesivamente hasta que el texto termina.

Por ejemplo, si dos usuarios S y B mandan respectivamente los mensajes m_S y m_B:

```
m_S: "Voten por mi, soy el mejor."

m_B: "No, no, mejor voten por mi."
```

entonces Introgram entrega el siguiente mensaje mezclado m_{SB}:

```
m_SB = "VoNoten, n poro, m mi, ejor soy elvoten mejor.por mi."
```

Escribe un programa que revierta los problemas que genera Introgram. En este programa el usuario ingresa un *string* mezclado m_{SB}, y el programa imprime en pantalla los *strings* originales m_S y m_B. Puede suponer que los *strings* m_S y m_B son **siempre del mismo largo**. El siguiente es un ejemplo de ejecución de su programa:

```
Introgram Corrector.
Ingrese string mezclado: VoNoten, n poro, m mi, ejor soy elvoten  mejor.por mi.
m_S: Voten por mi, soy el mejor.
m_B: No, no, mejor voten por mi.
```

Pregunta 2 (2/3)

La *Fonda Don Yadran* ha decidido implementar un sistema de atención a clientes que les permite, desde un terminal en cada mesa, realizar pedidos, pagar el consumo y hasta solicitar un móvil que les lleve de vuelta sanos y salvos a sus casas. Debes programar el flujo de atención del programa que permite realizar todas estas actividades.

Cada mesa de la fonda tendrá 3 clientes. En primer lugar se preguntará cuál cliente quiere pedir algún producto, ofreciéndole un menú con las opciones de consumo. Como la fonda gastó gran parte de su presupuesto en este sistema, el menú sólo ofrece empanadas, bebidas gaseosas y cervezas. El sistema debe permitir a los usuarios solicitar la cantidad que deseen de cada una de las opciones del menú. Los clientes pueden realizar más de un pedido.

Una vez que los clientes decidan solicitar la cuenta de la mesa, el sistema debe calcular el total de la cuenta, además de dar la opción de solicitar un móvil para retirarse. Como la *Fonda Don Yadran* cuida a sus clientes, en el caso que todos los clientes hayan consumido una Cerveza (o más), el sistema **obligatoriamente** les solicita un móvil, que es agregado de forma automática a la cuenta.

En la *Fonda Don Yadran* la comida nunca se termina, y siempre hay disponibilidad de móviles. Los precios son los siguientes: Empanada: $1000, Bebida: $500, Cerveza: $1500, Móvil: $10000.

A continuación se muestran dos ejemplos distintos de ejecución, puedes simplificar los strings cuanto estimes necesario:

```
Bienvenido a la fonda Don Yadran
¿Qué cliente desea ordenar? (0 para cuenta): 1
Elige tu comida: 1: Empanada, 2: Bebida, 3: Cerveza: 2
Ingresa cantidad: 2
¿Quieres otro producto? ("si"/"no"): si
Elige tu comida: 1: Empanada, 2: Bebida, 3: Cerveza: 3
Ingresa cantidad: 1
¿Quieres otro producto? ("si"/"no"): no
¿Qué cliente desea ordenar? (0 para cuenta): 0
¿Necesitas un móvil? ("si"/"no"): no
Total = $2500
```

```
Bienvenido a la fonda Don Yadran
¿Qué cliente desea ordenar? (0 para cuenta): 1
Elige tu comida: 1: Empanada, 2: Bebida, 3: Cerveza: 3
Ingresa cantidad: 1
¿Quieres otro producto? ("si"/"no"): no
¿Qué cliente desea ordenar? (0 para cuenta): 2
Elige tu comida: 1: Empanada, 2: Bebida, 3: Cerveza: 3
Ingresa cantidad: 1
¿Quieres otro producto? ("si"/"no"): no
¿Qué cliente desea ordenar? (0 para cuenta): 3
Elige tu comida: 1: Empanada, 2: Bebida, 3: Cerveza: 3
Ingresa cantidad: 1
¿Quieres otro producto? ("si"/"no"): no
¿Qué cliente desea ordenar? (0 para cuenta): 0
Te enviaremos un móvil.
Total = $14500
```

Pregunta 3 (3/3)

La flamante empresa *The YES Company*, está ingresando al mercado de la comida no convencional con productos novedosos: Chocolate que no engorda, carne vegetariana, agua en polvo, etc. Su fundador *Parim Kichara* necesita tu ayuda para gestionar y elegir los próximos productos que saldrán al mercado.

Tu trabajo es hacer un programa que dadas algunas condiciones revise si es posible generar una unidad de cierto producto. Considera que un producto está compuesto de distintos ingredientes con distintos costos asociados. A continuación se muestra un ejemplo usando *Mayo* como producto, donde se indica por cada ingrediente que la compone los gramos necesarios, costo por gramo y costo total.

Producto	Nombre Ingrediente	Índice	Gramos necesarios	Costo por gramo	Costo por Ingrediente
Mayo	Huevo	0	50	100	5000
	Aceite	1	10	250	2500
	Sal	2	1	1000	1000
	Jugo de Limón	3	5	300	1500

Cuadro 1: Información de Producción Mayo

Notamos que para producir una unidad de *Mayo* son necesarios 50 gramos de Huevo. Además el costo de cada gramo de huevo son \$100 pesos. Por lo que el costo total usado en huevo son $50 \cdot \$100 = \$5000$ pesos.

Para lograr tu cometido *The YES Company* te ha facilitado un módulo llamado `tyc` con las siguientes funciones que **ya han sido implementadas**:

- `num_ingredientes(p)`: Dado el nombre de un producto `p`, devuelve el número de ingredientes que lo componen. Por ejemplo, `num_ingredientes("Mayo")` retorna `4`.
- `gramos(p, i)`: Dado el nombre de un producto `p` y un índice de ingrediente `i` (los cuales van desde 0 hasta `num_ingredientes(p)` - 1), devuelve los gramos del ingrediente que se necesitan para producir una unidad de `p`. Por ejemplo `gramos("Mayo", 1)` retorna `10`, que son los gramos de aceite necesarios.
- `costo(p, i)`: Dado un producto `p` y un índice de ingrediente `i`, devuelve el costo del **ingrediente por gramo**. Por ejemplo `costo("Mayo", 2)` retorna `1000`, que es el costo de un gramo de sal.

El proceso de producción debe cumplir con que por cada producto, los gramos totales y costo total teniendo en cuenta todos los ingredientes no supere el máximo de lo que tienen presupuestado. Utilizando las funciones del módulo anterior deberás hacer lo siguiente:

a) Programar la función `gramos_totales(p)` que reciba el nombre de un producto `p` y retorne la suma en gramos de todos los ingredientes que lo componen. Por ejemplo, `gramos_totales("Mayo")` debería retornar 66 (Ya que $50 + 10 + 1 + 5 = 66$). **(1.5 puntos)**

b) Programar la función `costo_total(p)` que reciba el nombre de un producto `p` y retorne el costo total de producir una unidad del producto. Esto se calcula como la suma del costo total de cada ingrediente[1]. Además, se debe tener en cuenta la mano de obra de producción, lo cual significa sumar un 10 % del costo descrito anteriormente. Usando nuevamente como ejemplo a *Mayo*, el costo base se calcula como $5000 + 2500 + 1000 + 1500 = 10000$, luego sumando el 10 % de 10000, la función `costo_total("Mayo")` debería retornar 11000 **(2.5 puntos)**

c) Programar la función `es_valida(p, gramos_maximos, costo_maximo)` que reciba el nombre de un producto `p` y retorne `True` si es que la cantidad total en gramos de `p` no supera a `gramos_maximos` y el costo total no supera a `costo_maximo`. En caso contrario, debe retornar `False`. **(0.5 puntos)**

d) Escribir un programa en el cual debes preguntar al usuario por un producto, gramos máximos y costo máximo. El programa debe revisar si es válida la producción de una unidad de ese producto: en caso de que lo sea entregar un mensaje y termine la ejecución; en caso contrario, que vuelva a preguntar por los costos y cantidades máximas. **(1.5 puntos)** A continuación se muestra una ejecución de referencia, eres libre de cambiar la forma en que te comunicas con el usuario, siempre y cuando recibas y entregues la información relevante:

[1] Recuerde que el costo total de un ingrediente se calcula como los gramos necesarios multiplicado por el costo de cada gramo.

```
Ingrese Producto: Mayo
Ingrese Gramos Máximos: 60
Ingrese Costo Máximo: 12000
¡Configuración no válida!
Ingrese Gramos Máximos: 70
Ingrese Costo Máximo: 10400
Configuración no válida!
Ingrese Gramos Máximos: 100
Ingrese Costo Máximo: 11000
¡Configuración válida!
```