

Algoritmos y Programación II (75.41)

Trabajo Práctico N°1

17 de septiembre de 2012

1. Consigna

Los dueños de una famosa pizzería localizada en el corazón de Gerli están decididos a mejorar su imagen incorporando un programa que los ayude a gestionar los pedidos y el reparto de pizzas.

Como la pizzería es atendida por sus propios dueños, los pedidos se registran únicamente con el nombre del cliente (que identifica el pedido), la cantidad de pizzas (solo disponen de una variedad) y la zona de entrega. Por el momento la pizzería tiene predefinidas 5 zonas aunque planea agregar nuevas próximamente.

Si bien los clientes se comunican mayormente para quejarse por el servicio, también llaman para realizar modificaciones a los pedidos e incluso para cancelarlos. La pizzería sólo permite estas acciones si el pedido no ha sido preparado aún.

La pizzería necesita preparar los pedidos según la zona de envío y enviarlos en la única moto disponible para repartos. Por el momento la moto puede llevar hasta 5 pizzas sin que se enfríen (por esta limitación, no se registran pedidos de mas de 5 pizzas).

Utilizando los TADs vistos hasta el momento debemos realizar el programa para gestionar dichos pedidos y organizar el reparto.

1.1. Comportamiento del programa

El programa debe llevar un registro de los pedidos recibidos, permitiendo agregar un nuevo pedido, modificar la cantidad de pizzas de un pedido y cancelar un pedido.

Además debe contar con una opción para preparar todos los pedidos, agrupándolos según la zona de envío (los pedidos se preparan en el orden que fueron registrados). Una vez que los pedidos son preparados no pueden ser modificados ni cancelados.

Finalmente debe permitir enviar los pedidos a los clientes. Esto consiste en sacar algunos de los pedidos preparados por zona, utilizando una estrategia que permita optimizar los envíos. Los pedidos que no entren en la moto deberían postergarse para poder maximizar la cantidad de pedidos a enviar en cada viaje. El programa debe informar los pedidos enviados.

El programa debe permitir observar el estado de la pizzería incluyendo los pedidos registrados hasta el momento, los pedidos preparados (por zona) y los últimos N pedidos enviados (desde el más reciente al más antiguo).

El programa deberá contar con una menú de opciones para la funcionalidad mencionada.

1.2. Utilización de estructuras de datos

Para la realización de este trabajo es necesario utilizar las estructuras de datos vistas en clase (pilas, colas, listas enlazadas) además de crear una estructura *pedido*, con los datos indicados, para manipular los pedidos de pizza.

Todas las estructuras deben estar implementadas de la forma más genérica posible, correctamente documentadas, y con sus correspondientes pruebas unitarias.

2. Consignas opcionales

- La pizzería también es célebre por sus empanadas y quiere poder enviarlas junto con nuevas variedades de pizzas. El programa debería permitir que los pedidos incluyan una cantidad de empanadas y mantener una lista con al menos una variedad de pizza y una cantidad. Por

ejemplo: 1 Mozzarella, 2 Napolitana. La moto puede llevar, además de las 5 pizzas, hasta 2 docenas de empanadas en el canasto delantero.

- El programa debe permitir que la pizzería defina precios por pizza y por zona, los cuales serán pasados al programa por parámetro desde la consola, y permitir ver el total de ingresos por todos los pedidos enviados hasta el momento.
- Los pedidos se registran junto con la distancia desde la pizzería hasta el lugar de entrega. En este caso, los envíos deberían organizarse de manera que se muestren en orden según la distancia de entrega (primero el lugar más cercano y último el más lejano).
- Permitir que los pedidos de pizza se carguen desde un archivo, definiendo un formato adecuado, y mantener en otro archivo un historial con todos los envíos que se realicen.
- Implementar un intérprete para que el usuario controle el programa. El intérprete debe leer de *standard input* las acciones que el usuario quiera realizar. Por ejemplo, la siguiente orden podría registrar un nuevo pedido: **A Nombre 5 Z1**.

Para la aprobación del Trabajo Práctico se exige al menos una de las consignas opcionales. Además, todo el trabajo adicional será tenido en cuenta en la calificación final.

3. Criterios de aprobación

A continuación describimos criterios y lineamientos que deben respetarse en el desarrollo del trabajo:

3.1. Programa

El programa debe cumplir los siguientes requerimientos:

- Debe estar adecuadamente estructurado y modularizado, utilizando funciones definidas de la forma más genérica posible, sin caer en lo trivial.
- El código debe ser claro y legible.
- El código debe estar comentado y las funciones definidas, adecuadamente documentadas.

- El programa debe compilar sin mensajes de error (`-Wall -pedantic`), debe correr sin pérdidas de memoria, uso de valores sin inicializar, o errores en general. Es decir que, el programa debe correr en valgrind sin errores.
- Además, claro, debe satisfacer la especificación de la consigna.

3.2. Informe

El informe deberá consistir de las siguientes partes:

- Carátula con la información de los integrantes del grupo y el ayudante asignado.
- **Análisis, especificación y diseño:** describir el problema, cuál es la solución y cómo se lleva a cabo. Incluir las convenciones de código utilizadas (criterios para nombres de variables y funciones, criterios de indentación, etc).
- **Implementación:** Incluir aquí *todo* el código fuente utilizado en formato **monoespaciado**, para facilitar su lectura.
- **Pruebas unitarias:** probar las funcionalidades de cada estructura o porción del programa.
- Mantenimiento (*Opcional*): cambios importantes realizados a lo largo del trabajo.
- También *opcionalmente*, toda explicación adicional que consideren necesaria, referencias utilizadas, dificultades encontradas y conclusiones.

El informe debe estar lo más completo posible, con presentación y formato adecuados. Por ejemplo, este enunciado cumple con los requerimientos de un informe bien presentado.

4. Entrega

El trabajo consiste en:

- El informe impreso.
- El informe digital, en formato `.pdf`
- Una versión digital de los archivos de código fuente, separados del informe, en un archivo comprimido (`.zip` o `.tar.gz`).

Los dos últimos deben enviarse a la dirección `tps.7541rw@gmail.com`, colocando como asunto: **TP1 - Padrón1 - Apellido1 - Padrón2 - Apellido2**. El informe impreso debe entregarse en clase. El plazo de entrega vence el viernes 5 de Octubre de 2012.