



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC2233 Programación Avanzada (I/2015)

Tarea 3

1. Objetivos

- Utilizar los conceptos de la programación orientada a objetos y estructuras de datos para modelar un sistema.
- Aplicar conceptos de simulación basado en Eventos Discretos.
- Aprovechar las herramientas de SimPy3 para crear una simulación.

2. Problema

La mayoría de las veces se necesita una base de datos para poder extraer patrones y ver el comportamiento de los sistemas reales, sin embargo, el acceso a ésta es muy costoso y/o exclusivo. Por lo que generar información a partir de distribuciones de probabilidad reales nos permite tener un acercamiento a esta base de datos para poder estudiarla.

En esta tarea deberá crear un programa que se encargará de simular un supermercado, sus outputs serán utilizados para la búsqueda de perfiles de compradores en otros cursos del Departamento de Ciencia de la Computación. Para lograr lo anterior, primero deberá leer un par de archivos que contienen información acerca de las distintas entidades en el problema. Luego deberá realizar la simulación que consistirá en 2 partes, primero hacer una simulación basada en eventos discretos y luego agregar funcionalidades utilizando SimPy3.

Parte I

1. Especificaciones del Sistema

Deberá hacer una simulación del supermercado en base a las siguientes entidades:

1.1. Clientes

Los clientes se pueden dividir en distintos tipos, cada uno de estos tipos de ellos tiene distinta preferencia de productos y por lo tanto distintas probabilidades de compra. Además tienen asociado una probabilidad de aparición en el supermercado.

1.2. Lineas de Productos

Hay distintas líneas de productos, puede pensar esto como el pasillo de un supermercado.

1.3. Productos

Por cada línea de producto hay una lista de productos, cada uno posee un precio y cierta cantidad disponible.

1.4. Cajas

El supermercado posee una cantidad cajas, en estas se venden los productos a los clientes, se comportará como una cola que va recibiendo clientes que hacen compras.

1.5. Productos en caja

En la cola de la caja también se venden productos, sin embargo, las probabilidades de compra dependen del tiempo de espera que lleve un cliente.

1.6. Tiempo

La unidad de tiempo será en segundos, considere que todas las tasas y parámetros estarán basados en esta unidad. Debe tener en cuenta que el supermercado cierra cada 12 hrs (son 43200 seg), esto se considera un día (abre a las 9:00 am y cierra a las 9:00 pm). Luego de cerrar se atiende hasta que el último cliente sale del recinto.

2. Probabilidades y otras consideraciones

- Los tiempos de llegada distribuyen exponencial con cierta tasa.
- Cada cliente del tipo i posee un peso ap_i de aparición. La probabilidad de que un cliente llegue al supermercado es:

$$\frac{ap_i}{\sum_{j=1}^n ap_j}$$

- Cada cliente tiene una cierta probabilidad lp_i de ingresar a una cierta línea.
- Los productos de una línea tienen la misma probabilidad para todos los tipos de clientes, a menos que el cliente indique alguna preferencia.
- El tiempo de compra depende de cuanto compre y cuantas veces el cliente se haya cambiado de línea. Cada producto comprado agrega entre 5 a 10 seg y cada cambio de línea aumenta de 20 a 30 seg. Ambos valores distribuyen uniforme. El orden en que pasa por las líneas es aleatorio.
- El tiempo de atención en caja depende de la cantidad de productos que lleve el cliente, cada producto agrega 3 segundos. Además considere un tiempo base de demora extra de entre 0 a 1 minuto y 30 seg.
- Cada producto puede ser comprado hasta r veces por un cliente, el valor de r distribuye uniforme entre 1 y R .
- Cada tipo de cliente i llega con un monto máximo de gasto, que distribuye normal dado los parámetros μ_i y σ_i .
- El cliente elige una caja i con la siguiente probabilidad:

$$\frac{C - c_i}{C}$$

Donde c_i es la cantidad de personas en la cola de la caja i y C es la cantidad de personas en total haciendo cola entre todas las cajas. $C = \sum_{i=1}^n c_i$. Es decir, el cliente es más probable que elija la caja con menor cantidad de personas.

- Una vez que el cliente ya alcanzó su máximo de gasto o ya pasó por todas las líneas procede a ir a la caja.
- La probabilidad de que un cliente compre uno de los productos en caja es dada, pero va creciendo mientras más tiempo se esté en espera en un 1 % por cada segundo que pasa.
- Cuando el supermercado cierra, se debe atender a todos los clientes que estén dentro, pero se impide el ingreso a más personas.
- Los productos se reponen cada vez que el supermercado cierra.
- Todas las cajas tienen los mismos productos y la misma cantidad inicial. Esto no significa que todas las cajas compartan el mismo stock, piense cada caja como una línea separada en donde la persona solo puede comprar si es que ingreso a esa cola.

3. Carga de Datos

Junto con el enunciado encontrará con los siguientes archivos de texto.

■ clientes.txt

En la primera línea aparece cada cuantos segundos aparecen los clientes (Inverso de la tasa).

Luego cada tipo de cliente de la forma: (Cada tipo de cliente separado por \$

"nombre # peso-aparición # μ_i # σ_i # num-líneas # num-preferencia"

Después hay num-líneas por cada línea de producto a la que el cliente tiene probabilidad de ingresar:

"nombre-línea # probabilidad-ingreso"

Luego hay un carácter – separador.

Y después hay num-preferencia líneas por cada producto en el que tenga preferencia:

"nombre-producto # probabilidad-compra"

■ productos.txt

Una línea de producto separados por \$ de la forma:

"nombre-línea # num-productos"

Después de esto hay num-productos líneas por cada producto de la forma:

"nombre-producto # cantidad # precio # R # probabilidad-compra"

■ cajas.txt

En la primera línea aparece:

"num-cajas # num-productos "

Después de esto hay num-productos líneas por cada producto que se vende en caja de la forma:

"nombre-producto # cantidad # precio # R # probabilidad-compra"

Parte II

1. Especificaciones del Sistema

En esta parte se considerará el mismo proceso pero en un escenario más particular y definido: Los clientes preferenciales. No es necesario incluir a los otros tipos de clientes en esta parte.

1.1. Clientes preferenciales

Los clientes preferenciales son clientes que tienen más beneficios.

1.2. Estacionamiento

En general estos clientes llegan en vehículo, por lo que será necesario incluir un estacionamiento y tiempo de estacionamiento. El cupo por una butaca será determinado según la prioridad de urgencia de los clientes, es decir, si 2 clientes están esperando estacionamiento se asigna según prioridad. Considere que la prioridad de un cliente distribuye $\text{Uniforme}(0,3)$.

- La cantidad de estacionamientos debe ser un parámetro pedido al usuario de la simulación.
- El tiempo de estacionamiento distribuye Exponencial a tasa media 5 minutos.

1.3. Cajas

Los clientes preferenciales pueden querer realizar actividades poco frecuentes, por lo que las cajas pueden tener percances y demorarse más del tiempo normal debido al tipo de clientes. Se considerarán atascos de las cajas, por cada cliente el tiempo hasta un atasco distribuye $\text{Normal}(\text{media} = 10 \text{ minutos}, \text{desviación estándar} = 3 \text{ minutos})$. La cajera se demora en resolverlo un tiempo $\text{Uniforme}(4,6)$.

1.4. Políticas nuevas

Además, ahora el Supermercado cuenta con una política de atención al cliente donde aumentará las cajas disponibles para este tipo de clientes si es que todas las cajas están llenas con más de 3 personas en la cola. Cerrará una caja si luego de 30 minutos de haber abierto una hay menos de 3 personas en cualquiera de las otras, manteniendo siempre un mínimo de ciertas cajas disponibles.

Se debe repetir la menor cantidad de código posible, es decir, reutilizar lo que sea necesario de la parte I para la lectura de archivos, generación de carros de compra, etc. Y hacer con SimPy3 todo lo que esté al alcance de la librería, es decir, procesos de llegada, atención, etc.

2. Carga de Datos

Para esta parte deberá utilizar los mismos archivos anteriores, poniendo atención en que solo importará el tipo de cliente PREFERENCIAL. En la Parte I se debe omitir esa información y no usarla para la simulación.

Toda la demás información que no sea posible rescatar de los archivos *.txt* debe ser posible entregarla como input. Además puede considerar parámetros que el usuario de la simulación querrá cambiar y por lo tanto pedirlo como input, especificar estas decisiones en el README.md.

3. Funcionalidades

Para realizar la simulación deberá leer los archivos antes especificados, para esto se deberá pedir al usuario el tiempo que debe durar la simulación en días y si quiere correr la Parte I o la Parte II.

Puede asumir que los archivos tendrán los nombres ya indicados. El programa deberá entregar como output 3 archivos `.csv`:

- Un archivo que contenga todos los eventos ocurridos con sus tiempos correspondientes y características relevantes.
- Un archivo a modo de reporte general.
- Un archivo que muestre por cada línea:
Tipo cliente, producto1:cantidad_producto1:montototal_producto1, producto2:cantidad_producto2:...

Además debe darle la opción al usuario de mostrar en tiempo real los eventos a través de la consola.

3.1. Eventos

Todos los eventos deben mostrar el tiempo y día que fue hecho. Nos referimos a un evento como lo siguiente:

- Cuando un cliente ingresa al supermercado. En este caso se debe mostrar el tipo de cliente.
- Cuando un cliente termina de escoger los productos e ingresa a una caja. Debe mostrar a que caja ingreso y que productos lleva.
- La compra de un producto en caja. Debe indicar la caja, cliente y producto.
- La realización de la compra (abandono de la caja). Debe mostrar la caja, cliente y pago total.
- Si se agoto un producto. Debe mostrar el nombre del producto y la línea.
- El cierre del supermercado.

3.2. Reporte

El reporte debe ser generado independiente para la Parte I y para la Parte II, cada uno de estos deberá tener lo siguiente:

- Total ganado por el supermercado.
- Los 10 productos más vendidos por tipo de cliente.
- Gasto promedio por tipo de cliente.
- Tiempo promedio dentro del supermercado, por tipo de cliente.
- Tiempo medio de espera en la cola de acuerdo a la cantidad de cajas abiertas.
- Colas promedio de acuerdo a la cantidad de cajas abiertas.

Nota: En el README.md debe explicar brevemente cómo calculó cada medida de desempeño. Esto será parte de la evaluación.

4. Restricciones y alcances

- Tu programa debe ser desarrollado en Python 3.4
- Su código debe seguir la guía de estilos PEP8
- Si no se encuentra especificado en el enunciado, asuma que el uso de cualquier librería Python está prohibido. Pregunte por foro si se pueden usar librerías específicas.
- El ayudante puede castigar el puntaje¹ de tu tarea, si le parece adecuado. Se recomienda ordenar el código y ser lo más claro y eficiente posible en la creación algoritmos.
- La revisión de la tarea será realizada con distintos archivos `.txt`.
- Debe adjuntar un archivo *README.md* donde comente sus alcances y el funcionamiento de su sistema (*i.e.* manual de usuario) de forma *concisa* y *clara*.
- Cualquier aspecto no especificado queda a su criterio, siempre que no pase por encima de ningún otro.

5. Entrega

- **Output:** Entregue 3 archivos por cada parte corriendo cada modelo en un horizonte de 5 días.
- **Fecha/hora:** Martes 5 de Mayo de 2015 / 23:59.
- **Lugar:** GIT - Carpeta: Tareas/Tarea_03

Tareas que no cumplan con las restricciones señaladas en este enunciado tendrán la calificación mínima (1.0).

¹Hasta -5 décimas.