

Descripción

Este examen consta de dos ejercicios de 5 puntos cada uno

Simulador de Ascensor 5 puntos

Dimensionar el número de ascensores que deben instalarse en edificios de gran tráfico no es sencillo. Para facilitar esta tarea se propone simular el comportamiento de un ascensor y determinar con ello la capacidad de atender la demanda bajo distintos escenarios. El ascensor es un mecanismo capaz de albergar simultáneamente un número determinado de personas para trasladarlas entre diferentes plantas. El ascensor atiende n plantas entre las cuales debe desplazarse bajo demanda.

El ascensor constituye, en este contexto, un recurso común con capacidad limitada y con estado interno, que por simplificar el problema, limitaremos a dos estados: LIBRE y OCUPADO.

Las personas que desean utilizar el ascensor se ejecutan de forma concurrente y su ciclo de vida es el siguiente: Llamam al ascensor y se quedan a la espera hasta que este llega a la planta. Si logran subir (puede haber más personas esperando en la misma planta que se suban antes), el ascensor les llevará a su destino, en caso contrario la persona tendrá que volver a llamar al ascensor.

De nuevo, por simplificar, el ascensor solo tendrá capacidad para llevar a una persona simultáneamente.

Bajo este escenario diseñe y sincronice adecuadamente:

a) El recurso compartido “Ascensor” con al menos un constructor que permita inicializar los atributos (número de plantas, capacidad de personas, planta inicial, etc.) y con los métodos que permitan explotar sus servicios (llamar al ascensor para que si está libre se dirija a una planta, subirse al ascensor para ir a una planta, etc.). Se recuerda que para simplificar la dificultad del problema, el ascensor sólo tendrá capacidad para trasladar a una persona a la vez.

b) La clase que representa a las personas, de manera que su ejecución será concurrente con el resto de personas. El constructor de cada persona la sitúa al azar en una planta con la intención de acudir al azar a otra planta. La persona llamará al ascensor y tendrá que seguir el ciclo de vida que se ha descrito en el enunciado.

c) La aplicación principal responsable de crear un objeto que represente un ascensor en un edificio de n plantas y con capacidad para m personas (por simplificar m será 1). La aplicación creará también una lista de personas cada que se lanzarán para su ejecución concurrente. La aplicación generará personas hasta un máximo de q personas. La aplicación finalizará cuando se hayan atendido a esas q personas y mostrará el tiempo medio de espera de las personas.

Servicio generador de contraseñas 5 puntos

Objetivo: Crear un servidor TCP en Java que genere contraseñas aleatorias a partir de una cadena de entrada sin espacios. El programa recibirá como parámetro el puerto de escucha en el momento de su ejecución.

Inicio del programa: El programa se iniciará mediante línea de comandos, y deberá recibir como parámetro el puerto de escucha. Por ejemplo, si el puerto es 5000, la línea de comandos para ejecutar el programa podría ser:

> **java PasswordGenerator 5000**

Servicio TCP: Implemente un servicio que escuche en el puerto especificado y acepte conexiones de clientes. Debe ser capaz de recibir una cadena de texto sin espacios a través de esta conexión TCP.

Generación de contraseñas: A partir de la cadena de entrada, el programa generará 10 posibles contraseñas. Para ello:

- Cada carácter se podrá escribir tanto en mayúscula como en minúscula de forma aleatoria con una probabilidad del 50%. (si número aleatorio entre 0 y 1 es menor que 0.5; Cuidado con los números mágicos)
- Se deben realizar las siguientes transformaciones de caracteres: la 'o' será 0, la 'i' será 1, la 'a' será 4, la 'e' será 3 y la 't' será 7. También con una probabilidad del 50%.
- Se añadirán 2 caracteres especiales de entre el listado "\$%&()-_" al final de cada contraseña.

Envío de contraseñas: Una vez generadas las contraseñas, estas se enviarán de vuelta al cliente a través de la conexión TCP.

En la página anexa puedes ver ejemplos.

Evaluación:

Paso 1 – Inicio del programa (1 punto): Se evaluará si el programa recibe correctamente el puerto de escucha como parámetro y se inicia adecuadamente.

Paso 2 – Servicio TCP (1 puntos): Se evaluará la correcta implementación del servicio TCP que escuche en el puerto especificado.

Paso 3 – Generación de contraseñas (2 puntos): Se evaluará si el programa genera correctamente las contraseñas basadas en las reglas especificadas.

Paso 4 – Envío de contraseñas (1 puntos): Se evaluará si el programa envía correctamente las contraseñas generadas al cliente.

Total – 5 puntos

Buena suerte con la implementación!

Entrada	Posibles Salidas
"hola"	"H0L4*\$"
	"h0L4%&"
	"h0l4(-"
	"H0LA_%"
	"h0LA-*"
	"H0La-&"
	"h0La_%"
	"h0lA*-"
	"H0lA*%"
	"H0LA*_ "
"java"	"JAv4*%"
	"JaV4-*"
	"JAv4_\$"
	"jAv4*%"
	"JaV4\$%"
	"jAV4-*"
	"jav4\$%"
	"JaV4_*"
	"Jav4*_ "
	"jAv4-\$"

Recuerda que estos son solo ejemplos de salidas, y los resultados específicos pueden variar debido a la naturaleza aleatoria de la generación de contraseñas.