Universidad Central de Venezuela
Facultad de Ciencias
Escuela de Computación
Lenguajes de Programación (6204)
Semestre 1-2025

Proyecto # 3 – Programación Concurrente
The Bucket Theory

En la fiesta de fin de semestre, se planea realizar una cervezada en la que los estudiantes de la materia reutilizarán el sistema de tres barriles comunicantes, previamente trabajado en los proyectos de programación funcional y lógica. Se requiere implementar una solución concurrente para este problema, utilizando monitores e hilos (Threads) en Java.

Se debe tener en consideración lo siguiente:

1. **Barriles:** se disponen de tres barriles, denominados “A”, “B” y “C”, que cuentan con las mismas condiciones y restricciones que en los proyectos anteriores.
2. **Estudiantes:** cada estudiante está definido por su nombre, edad y la cantidad de tickets que tiene para retirar vasos de cerveza.
3. **Proveedores:** son las personas responsables de agregar más cerveza a los barriles cuando sea necesario.

La interacción debe ser a través de datos de entrada. Al ejecutar el programa debe suministrarse un archivo de entrada en formato TXT (el nombre del mismo se recibirá como argumento en el método main), ejecutar su lectura en batch y disparar los hilos correspondientes. El archivo debe seguir el siguiente formato:

```
A, 10, 10
B, 7, 0
C, 3, 0
Estudiantes, 34
Proveedores, 3
```

Las primeras tres líneas del archivo describen **el estado inicial de los barriles**, incluyendo su identificador, capacidad y cantidad de cerveza. La cuarta línea indica **el número de estudiantes que participarán en la fiesta**, mientras que la última línea especifica **la cantidad de proveedores de cerveza presentes**.

Se espera que ud. implemente lo siguiente:

1. **Proveedor de cerveza.** La simulación debe incluir la implementación de un “productor”, responsable de recargar los barriles cuando al menos uno tenga capacidad disponible. De existir perdida de cerveza por desborde, debe ser registrada y reportada en su totalidad al finalizar la ejecución.
2. **Consumidor.** Los estudiantes mayores de edad, pueden solicitar servirse una o varias cervezas, según la cantidad de tickets que tengan disponibles en ese momento. Si el barril no tiene la cantidad total solicitada, se le servirá lo que esté disponible y deberá esperar hasta que se reabastezca con más cerveza. Cuando se agoten los tickets, el estudiante deberá retirarse de la fiesta. Si aún le quedan tickets, podrá seguir disfrutando y solicitando cervezas según lo acordado.

Consideraciones para la entrega

- Realice las validaciones que sean necesarias.
- Ud. debe identificar los procesos, recursos críticos (junto con las operaciones de acceso) y las condiciones de sincronización presentes en el problema. Esto lo debe documentar en un archivo en formato **markdown** (README.md).
- Documente también las decisiones de diseño que tome para solucionar el problema e indique las instrucciones de uso.
- No está permitido utilizar soluciones de terceros (esto incluye soluciones generadas por LLMs). Cualquier material consultado para ideas de solución o documentación debe ser referenciado en una sección del **README.md**, indicando para qué sirvió.
- El proyecto puede realizarse en parejas y la fecha de entrega será el martes **08/07/2025**, hasta las 11:59 (VET).
- El proyecto se debe entregar en un archivo comprimido cuyo nombre debe seguir la siguiente sintaxis: <Cedula1>-<Apellido1>-<Cedula2>-<Apellido2>-Proyecto_3.[rar|zip]. El archivo ZIP/RAR debe contener una carpeta con el mismo nombre del archivo y debe incluir: el archivo README.md, los casos de prueba, el proyecto de código fuente con la implementación de la solución y un archivo makefile o similar para compilar la aplicación **en cualquier plataforma**.
- La entrega debe compilar en una consola/terminal y NO tener dependencias del IDE utilizado.
- El proyecto se debe enviar por e-mail a ldpucv@gmail.com colocando como asunto el nombre del archivo comprimido.

José Yvimas, Junio 2025