GENERALITAT VALENCIANA
Conselleria d'Educació, Cultura i Esport

ceedcv CENTRE ESPECÍFIC D'EDUCACIÓ A DISTÀNCIA DE LA COMUNITAT VALENCIANA

EXAMEN PRÁCTICO

Primer trimestre – TEMA 1

Licencia

1. Evaluación

1.1. Validación del código

Cuando acabes el examen debes acceder a un segundo enlace proporcionado por el profesor para realizar la validación del examen, que constará de varias preguntas y posibles cambios sobre tu código.

1.2. Infracciones que conllevan el suspenso del examen

Posibles infracciones:

- Se ha copiado en el examen.
- El alumno no es capaz de contestar a preguntas relacionadas con su examen.
- No se realiza la validación del examen.
- El uso de inteligencia artificial para la resolución.
- El programa no lleva el nombre del alumno/a:
  - Si me llamo Miguel Oliver, mi entrega se llamará: Examen_miguel_oliver.py

1.3. Puntuación

Si tu programa no se ejecuta correctamente, la nota máxima será un 6. Soluciones válidas pero que no sean eficientes o las descritas en el enunciado, restarán hasta 5 puntos del examen.

Se valorará la programación modular, la reutilización de funciones, etc.

Como mínimo se deberá imprimir por pantalla la misma información que se pide en el enunciado.

Si el código no incluye las funcionalidades que se piden en el enunciado, la nota máxima será un 6.

Será obligatorio el control de excepciones, si no se implementa correctamente la nota máxima será de 6.

2. Ejercicio 1

Deberás implementar un sistema multihilo que simule una fábrica, un almacén y varias tiendas, gestionando eficientemente la producción y distribución de productos mediante hilos personalizados con clases derivadas de `Thread`. Se deben utilizar las mínimas variables globales y utilizar los argumentos.

Especificaciones

1. **Máquinas de Producción (Fábrica)**
   - **3 máquinas**, cada una representada por un hilo personalizado con nombres `Fabrica1`, `Fabrica2` y `Fabrica3`.
   - Cada máquina puede producir **5 tipos de productos**: `A`, `B`, `C`, `D`, `E`.
   - **Tiempos de configuración** para cambiar entre productos (segundos):

```
tiempos_configuracion = [0.2, 0.4, 0.3, 0.5, 0.6]  # A, B, C, D, E
```

   - **Tiempos de producción por unidad** (segundos):

```
tiempos_produccion = [0.3, 0.35, 0.2, 0.5, 0.4] # A, B, C, D, E
```

2. **Almacén**
   - **Capacidad máxima de stock** por producto:

```
capacidad_maxima = [20, 19, 16, 22, 24] # A, B, C, D, E
```

   - **Stock inicial** de cada producto:

```
stock_inicial = [15, 14, 12, 16, 18] # A, B, C, D, E
```

   - La cantidad a producir de cada producto dependerá de lo solicitado por las tiendas desde el último pedido.
   - El almacén inicia producción cuando el stock cae por debajo de un porcentaje que debes definir y justificar.

3. **Tiendas**
   - **10 tiendas**, cada una representada por un hilo personalizado con nombres `Tienda1`, `Tienda2`, ..., `Tienda10`.
   - **Intervalos de pedidos** (segundos):

```
intervalos_pedidos = [10, 12, 8, 15, 18, 9, 11, 14, 13, 12]
```

   - **Cantidad de productos solicitados**: Cada pedido tiene una cantidad aleatoria entre 0 y 4 unidades por producto.

4. **Transporte**

- **Flota disponible**: Hay una flota de 2 **camiones**, cada uno representado por un hilo personalizado: **Camión1 y Camión2.**
- **Tiempos de carga y descarga**: La carga y la descarga de productos en un camión toma un tiempo de **0.25 segundos por unidad cargada o descargada**.
- **Restricción de destinos**: Un camión puede entregar productos a **un máximo de 4 tiendas por viaje**.
- **Capacidad de carga**: Cada camión puede transportar hasta **20 unidades** de producto en total, repartidas entre diferentes tipos de productos.
- **Tiempos de transporte** entre el almacén y las tiendas (segundos), también supondremos que entre tiendas se tarda un tiempo similar al que hay entre almacén y tienda:

```
tiempos_transporte_tienda = [2, 3, 1.5, 2.5, 2, 3.5, 4, 4.5, 3.2, 4.1]
```

- El transporte de productos desde la fábrica al almacén será instantáneo (simulado como cinta).

5. **Gestión de Prioridades**

- Los pedidos se gestionan en una cola **FIFO con prioridades**:
  - Los pedidos pueden ser **prioritarios** o **normales** (determinados aleatoriamente).
  - Tras procesar un pedido prioritario, deben atenderse **2 pedidos normales** antes de procesar el siguiente prioritario.
- Los pedidos de las tiendas se deben enviar completos en cada camión. No se pueden dividir en varios envíos.
- **EXTRA 1 punto:** Se rellenará el camión según la cola de **FIFO con prioridades**, si un pedido no cabe, se debe buscar en la cola el siguiente pedido que pueda ser enviado y rellenar el camión.
- **SIN EXTRA:** no es necesario rellenar el camión, añadiremos pedidos hasta que podamos.

6. **Planificación de la Producción**

- Implementa los algoritmos **FIFO** y **Shortest Job First (SJF)** para planificar la producción en las máquinas.

Información a Mostrar

1. **Producción en la Fábrica:**

- Fin de la producción de cada producto con el nombre de la máquina y la cantidad producida. Ejemplo: `Fabrica1 ha producido 10 unidades de A.`
- Cambios de configuración entre productos. Ejemplo: `Fabrica2 configurada para B.`

2. **Almacén:**
    - Nivel de stock antes y después de un pedido. Ejemplo: `Almacén envió 3 unidades de C a Tienda3. Stock restante de C: 117.`
    - Activación de producción debido a bajo stock. Ejemplo: `Almacén solicitó producción de 10 unidades de A.`
3. **Tiendas:**
    - Pedidos realizados por las tiendas con sus prioridades. Ejemplo: `Tienda7 solicitó 2 unidades de D con prioridad alta.`
4. **Transporte:**
    - Inicio y fin de transporte hacia las tiendas. Ejemplo: `Transporte completó entrega a Tienda4.`

Puntuación

1. Programación modular y estructura básica del programa con la herencia y la creación de los hilos. **(2 puntos)**
2. Control de excepciones. **(1 punto)**
3. Información por pantalla. **(1 punto)**
4. Control de concurrencia. **(1 punto)**
5. Desarrollar el FIFO con prioridades. **(2 puntos)**
6. Programar las funciones para rellenar los camiones. **(1 punto EXTRA)**
7. Implementar el FIFO y SJF en las fábrica. **(2 puntos)**
8. Responder a las siguientes preguntas: **(1 punto)**
    1. Evalúa cuál de los algoritmos de las máquinas es más eficiente en términos de producción.
    2. Define el porcentaje mínimo de stock para activar la producción. ¿Por qué elegiste ese valor? Analiza su impacto en el rendimiento del sistema.