

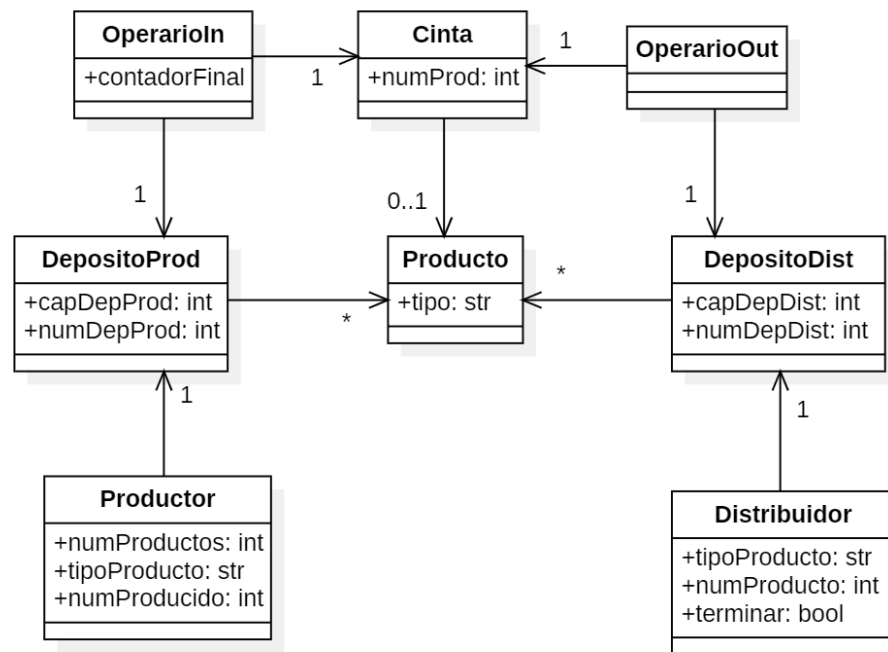
Caso 1 - Infraestructura Computacional

Nombres y códigos	1
Arquitectura general de la solución	1
Estrategias para los procesos	2
Implementación y pruebas	3
Sincronización y funcionamiento global del sistema	4

Nombres y códigos

- Diego Alejandro Rodriguez - 202225217
- Carlos Andrés Peña - 20222516
- Isabella Sarquis Buitrago - 202221542

Arquitectura general de la solución



Explicando el diseño (diagrama de clase) y funcionamiento del programa

Para la solución, se planteó un diagrama similar al que se muestra en el enunciado, a continuación se describe el funcionamiento del programa y sus clases:

Productor: El productor es el que produce la cantidad de productos que se pidan por consola, este meterá los productos producidos al depósito de producción cada que haya espacio disponible.

DepositoProd: Es el buffer que va a recibir los recursos del productor antes de ser extraídos por un operario interno.

OperarioIn: Es el operario que traspasa los productos del depósito de producción a la cinta, sólo puede mover uno a la vez y solo puede poner uno en la cinta a la vez.

Cinta: Es la cinta de producción, pasa los productos del OperarioIn al OperarioOut, solo puede haber 1 producto pasando por ella a la vez.

OperarioOut: Saca los productos de la cinta para dejarlos en el Depósito de distribución si es que hay espacio.

DepositoDist: Aquí se almacenan los recursos (productos) que salen de la cinta y son depositados aquí por el OperarioOut, tiene una capacidad limitada.

Distribuidor: Saca los productos del depósito de distribución, liberando el espacio.

Producto: Es el eje central del diseño, pasan por cada una de las demás clases en algún momento.

Estrategias para los procesos

Desde el main se pide el número a producir por Operario y la capacidad de los depósitos de producción y distribución. Luego se crean 2 productores y 2 distribuidores por cada tipo de producto (A y B) y se crean los operarios In y Out. Se inician los threads para cada una de las instancias creadas, para dar paso a los siguientes procesos:

- Producción
 - Para la producción, vamos a producir cuantos productos pueda cada productor, siempre y cuando el depósito de producción no esté lleno. Cuando este se llena, se inicia una espera pasiva, que termina con la notificación por parte del OperarioIn.
- Transporte
 - Hacia la cinta
 - Para el transporte hacia la cinta, vamos a usar una estrategia de espera semiactiva. Esto se debe a que en el momento en el que OperarioIn no tenga productos de parte del depósito de producción, o la Cinta se encuentre llena, este va a esperar revisando continuamente para continuar con su ejecución.
 - Fuera de la cinta
 - Para el transporte fuera de la cinta, vamos a usar la estrategia de espera semiactiva. Esto se debe a que en el momento en el que OperarioOut no tenga productos de parte de la Cinta, o el depósito de distribución esté lleno, este va a esperar revisando continuamente para continuar con su ejecución.

- Distribución
 - Para la distribución, vamos a avisar cada vez que llenemos el depósito a el tipo de thread (Distribuidor) correspondiente, para que salga de su espera pasiva. En caso de que no tengamos productos para distribuir, el Distribuidor espera.

Implementación y pruebas

Para validar el trabajo realizado implementamos varias herramientas. Principalmente por medio de prints en consola podemos ver como avanza el sistema.

Los siguientes son ejemplos de estas herramientas implementadas:

- Productor:
 - Se imprime cada vez que se inicia uno con su identificador único, y su meta de producción con su tipo asignado.
 - Se imprime cuando inicia y termina su ejecución.
- Operarios:
 - Se imprime cuando inician y terminan su ejecución.
- Distribuidor:
 - Se imprime cada vez que se inicia uno con su identificador único, y tipo de producto a recibir asignado.
 - Se imprime cuando inicia su ejecución.
 - Por último, cuando termina su ejecución se imprime un mensaje con el identificador del distribuidor, su tipo de producto y cuántos de los mismos obtuvo.

Por lo anterior, es en el último mensaje (el de finalización del distribuidor), en el que se puede verificar que la cantidad de productos producidos es igual al de distribuidos. La forma en que nos aseguramos de que este último mensaje sea impreso de manera correcta es que solo ocurre cuando el distribuidor recibe un producto de tipo "FIN_[A/B]".

A continuación se adjunta captura de cómo los prints llevan el estado del proceso:

```

Ingrese el número de productos a producir por cada Operario: 4
Ingrese la capacidad del deposito de producción: 4
Ingrese la capacidad del deposito de distribución: 4
Productor 0 creado meta de 4 productos: A
Productor 1 creado meta de 4 productos: A
Productor 2 creado meta de 4 productos: B
Productor 3 creado meta de 4 productos: B
Distribuido 0 creado, tipo de producto: A
Distribuido 1 creado, tipo de producto: A
Distribuido 2 creado, tipo de producto: B
Distribuido 3 creado, tipo de producto: B
OperarioIn creado
OperarioOut creado
OperarioIn inicia
OperarioOut inicia
Distribuidor 0 inicia
Distribuidor 3 inicia
Distribuidor 2 inicia
Productor 3 inicia
Productor 1 inicia
Productor 0 inicia
Productor 2 inicia
Distribuidor 1 inicia
Productor 2 termina
Distribuidor 3 termina con 2 productos de tipo B
Productor 0 termina
Productor 3 termina
Productor 1 termina
Distribuidor 1 termina con 8 productos de tipo A
OperarioIn termina
OperarioOut termina
Distribuidor 2 termina con 8 productos de tipo B
Distribuidor 0 termina con 2 productos de tipo A

```

Sincronización y funcionamiento global del sistema

Continuando, se va a exponer cómo se realiza la sincronización para cada pareja de objetos:

Productor-DepositoProd: El productor tiene la tarea de producir el número de productos que sean ingresados por consola y almacenarlos en el deposito de producción, no obstante, el depósito de producción tiene una capacidad limitada por lo que actúa como un buffer almacenando la cantidad de productos posibles entre el productor y el OperarioIn, en esta relación se implementa espera pasiva donde el productor espera a ser notificado cuando haya espacio.

DepositoProd-Producto: Un depósito de producción almacena la cantidad de productos que sean producidos por el productor siempre y cuando tenga espacio disponible. Estos son extraídos por el operarioIn antes de pasarlos a la cinta. DepositoProd actúa como un buffer para ambos el Productor y el OperarioIn, asegurando que ninguno de los dos manipule el ArrayList “productos” donde se almacenan los productos al mismo tiempo, asegurando así la integridad del programa.

DepositoProd-OperarioIn: El operario interno se encarga de extraer los productos del depósito para dejarlos en la cinta, este implementa espera semi-activa usando yield() para ceder su quantum de tiempo en caso de que la cinta esté ocupada o no haya productos en el deposito de producción.

OperarioIn-Cinta: El operario debe dejar los productos extraídos en la cinta mientras esta no esté ocupada. Usa espera semi-activa en caso de que esté ocupada la cinta.

Cinta-Producto: Una cinta envía un producto entre el operarioIn y el operarioOut, solo puede haber 1 producto en la cinta a la vez. La cinta actúa como un buffer de tamaño 1. Esto garantiza que la variable sobre la cual se almacena el producto (el arreglo de tamaño 1 “productoEnCinta”), sea sincronizada entre el OperarioIn y el OperarioOut.

Cinta-OperarioOut: El operarioOut extrae el producto que se encuentre en la cinta para depositarlo en el depósito de distribución, usa espera semi-activa en caso de que no hayan productos en la cinta.

OperarioOut-DepositoDist: El operario deposita los productos que tome de la cinta para almacenarlos acá siempre que haya espacio, si no hay espacio, usa espera semi-activa hasta que el depósito libere recursos para almacenarlo.

DepositoDist-Producto: Un depósito de distribución tiene una capacidad máxima para almacenar objetos del tipo producto, actúa como buffer entre el OperarioOut y los distribuidores.

DepositoDist-Distribuidor: Los distribuidores consumen los productos que se encuentren en el buffer (depósito) para liberar el espacio que estos ocupaban y dando paso a nuevos productos de parte del OperarioOut.