

Programación 3

Trabajo Práctico Especial



Cursada 2025

Contexto

En una fábrica de construcción de autopartes se dispone de un conjunto de máquinas capaces de producir un número determinado de piezas. Cada máquina produce un número establecido de piezas una vez que se pone en funcionamiento, esta cantidad de piezas se conoce de antemano y la máquina no puede detener su funcionamiento hasta completar esa cantidad. Por una cuestión de ahorro energético sólo una máquina puede estar funcionando a la vez. Por otro lado, no hay restricciones respecto a volver a poner en funcionamiento una máquina que ya ha estado en funcionamiento anteriormente.

El trabajo práctico de cursada propone resolver un problema, dentro de este contexto, mediante dos técnicas de programación: Backtracking y Greedy.

Enunciado

Dado un número determinado de piezas a producir se desea encontrar, de existir, la secuencia de máquinas que se deben usar para **minimizar la cantidad de puestas en funcionamiento totales** requeridas para completar la producción.

Por ejemplo, suponiendo las siguientes máquinas con la cantidad de piezas que produce cada una: (M1, 7), (M2, 3), (M3, 4), (M4, 1)

Si necesitamos producir 12 piezas una configuración óptima posible sería [M1 - M3 - M4]. También se podría haber propuesto una secuencia [M3 - M3 - M3]

Aplicación

Se debe construir una aplicación que comience levantando a memoria una configuración de máquinas, piezas que producen y piezas totales a producir de un archivo de texto.

El archivo de texto deberá respetar el siguiente formato:

PiezasTotales

Máquina1,Piezas

...

MáquinaN,Piezas

Para el caso anterior de ejemplo, se puede configurar un archivo como el siguiente:

12

M1,7

M2,3

M3,4

M4,1

Luego se deberá resolver el problema mediante las técnicas de Backtracking y Greedy.
Por último, se deberán presentar los resultados teniendo en cuenta distintas métricas que permitan visualizar, mínimamente, la calidad de la solución y el costo de obtener dicha solución, con ambas técnicas.

Para la presentación de resultados se propone el siguiente formato:

Backtracking

Solución obtenida: secuencia de máquinas.

Solución obtenida: cantidad de piezas producidas y cantidad de puestas en funcionamiento requeridas.

Métrica para analizar el costo de la solución (cantidad de estados generados)

Greedy

Solución obtenida: secuencia de máquinas.

Solución obtenida: cantidad de piezas producidas y cantidad de puestas en funcionamiento requeridas.

Métrica para analizar el costo de la solución (cantidad de candidatos considerados)

También se solicita una breve explicación de la estrategia elegida para resolver el problema, mediante cada técnica solicitada. Para esto, se propone acompañar la implementación de la técnica con un breve comentario antes de la función principal, por ejemplo:

```
/*
 * <<Breve explicación de la estrategia de resolución. Por ejemplo:
 *   - Cómo se genera el árbol de exploración.
 *   - Cuáles son los estados finales y estados solución.
 *   - Posibles podas.
 *   - etc.>>
 */
public Solucion backtracking(...) {}

/*
 * <<Breve explicación de la estrategia de resolución. Por ejemplo:
 *   - Cuáles son los candidatos.
 *   - Estrategia de selección de candidatos.
 *   - Consideraciones respecto a encontrar o no solución.
 *   - etc.>>
 */
public Solucion greedy(...) {}
```

Requisitos de la entrega

Se deberá entregar un proyecto (junto al código fuente) que compile correctamente el código de la aplicación solicitada. La entrega se realizará por email a la cuenta de la materia, pudiendo adjuntar en un archivo comprimido sólo el código fuente o compartiendo un link a un repositorio de git.

La fecha de entrega se encuentra disponible en el aula virtual de la materia.

Importante: no se aceptarán entregas fuera de término.