



IIC2115 – Programación como Herramienta para la Ingeniería (I/2018)

Actividad 8

Objetivos

- Aplicar contenidos de algoritmos y complejidad

Entrega

- **Lenguaje a utilizar:** Python
- **Lugar:** GitHub
- **Hora:** 16:55
- **Desarrollo:** Individual

Problema 1

Bastían tiene una caja de pasteles cuadrada donde caben $n = m \times m$ pasteles. El estado de cada uno de estos puede ser podrido (representado con una P) o comestible (representado por una C). En un día, los pasteles podridos hacen que todos sus vecinos sanos se pudran, y esto sigue hasta que eventualmente todos los pasteles de la caja se pudran. Dado un conjunto de $n = m \times m$ estados ingresados por el usuario (uno por cada pastel de la caja), encuentra el número de días en que todos los pasteles de la caja se pudren.

Ejemplo

Input: C C C C P C C C C #Caja con $3 \times 3 = 9$ pasteles, donde el elemento central está podrido

Output: 1 # En un día todos se pudren

Restricción: Considere que en el estado inicial, más de un pastel puede estar podrido. Su solución debe tener complejidad menor a $\mathcal{O}(n^2)$, en donde n representa el número total de pasteles del problema. No está permitido usar una simulación para solucionar el problema, e ir calculando día por día iterativamente.

Problema 2

Hugo tiene un árbol de navidad decorado con vistosos adornos. Como todos saben, a Hugo le encanta programar, por lo que su árbol de navidad tiene la estructura de un árbol binario. Se cumple que en cada nodo de su árbol hay un adorno, y cada adorno tiene un determinado valor. Como buen computín, Hugo quiere tener el árbol más pequeño posible, pero que al mismo tiempo tenga el mayor valor posible. Hugo

decide entonces eliminar las k ramas más baratas del árbol (desde un nodo hoja hasta el nodo raíz). Ayude a Hugo a encontrar estas ramas.

Junto al enunciado hay un archivo llamado **arboldehugo.txt**, que contiene la información del árbol de la siguiente forma:

- La primera línea contiene el número de nodos del árbol (estos están enumerados de 1 a N , donde N es el número de nodos).
- Las siguientes N líneas contienen información sobre los hijos de cada nodo. Cada una de estas líneas contiene el número del nodo, y a sus hijos junto a su peso, de la forma $a; b$, en donde a es el número del nodo y b es su peso. Además, la información entre nodos hijos estarán separados por un espacio. Un ejemplo de línea sería el siguiente:

1 2;4 3;4

Deberá imprimir en consola, para cada rama, la ruta desde la raíz, indicando el número de cada nodo y el valor total de la rama. El orden de impresión de las ramas debe ser creciente en función de su valor.

Restricción

Para resolver este problema utilice backtracking. Su solución debe tener complejidad menor a $O(k \cdot n^2)$, en donde n representa el número de nodos del problema y k la cantidad de ramas a seleccionar. Considere finalmente que el árbol descrito en el archivo es sólo un ejemplo, por lo que su algoritmo debe funcionar correctamente con cualquier árbol binario.

Política de Integridad Académica

Los alumnos de la Escuela de Ingeniería deben mantener un comportamiento acorde al Código de Honor de la Universidad:

“Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a respetar los principios y normativas que la rigen. Asimismo, prometo actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad y en la realización de todo trabajo, particularmente en aquellas actividades vinculadas a la docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas y cuidaré los bienes de la Universidad.”

En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un procedimiento sumario. Ejemplos de actos deshonestos son la copia, el uso de material o equipos no permitidos en las evaluaciones, el plagio, o la falsificación de identidad, entre otros. Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica en relación a copia y plagio: Todo trabajo presentado por un alumno (grupo) para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno (grupo), sin apoyo en material de terceros. Si un alumno (grupo) copia un trabajo, se le calificará con nota 1.0 en dicha evaluación y dependiendo de la gravedad de sus acciones podrá tener un 1.0 en todo ese ítem de evaluaciones o un 1.1 en el curso. Además, los antecedentes serán enviados a la Dirección de Docencia de la Escuela de Ingeniería para evaluar posteriores sanciones en conjunto con la Universidad, las que pueden incluir un procedimiento sumario. Por “copia” o “plagio” se entiende incluir en el trabajo presentado como propio, partes desarrolladas por otra persona. Está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la cita correspondiente.