

Programa 1 - Adoquinamiento

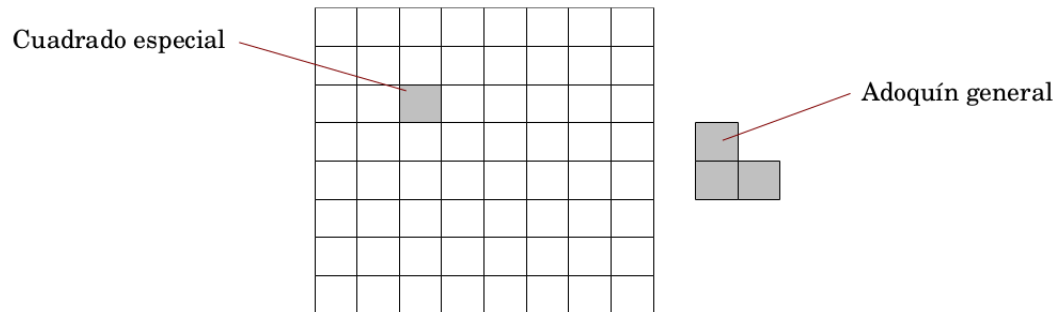
Análisis de algoritmos

Peto Gutiérrez Emmanuel
María de Luz Gasca Soto

19 de agosto de 2019

1. Introducción

El problema del adoquinamiento consiste en pavimentar una región dividida en cuadrados iguales. La región tiene un área de $m \times m$ cuadrados. A partir del cuadrado especial (el cual se coloca arbitrariamente en el área), se pavimenta con el adoquín formado por tres cuadrados en forma de L.



2. Planteamiento del problema

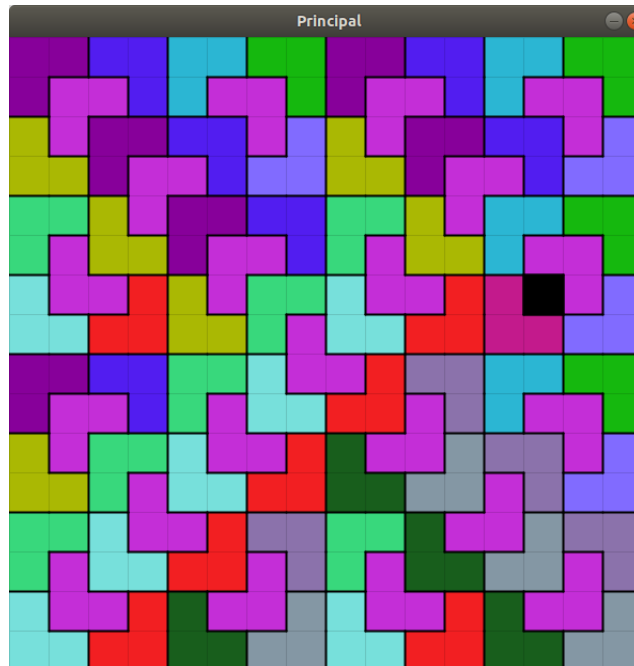
Sea m una potencia de 2, adoquinar la región de $m \times m$ con el adoquín dado, cubriendo cada cuadrado exactamente una vez, con excepción del cuadrado especial, el cual no será cubierto por ningún adoquín.

3. Implementación

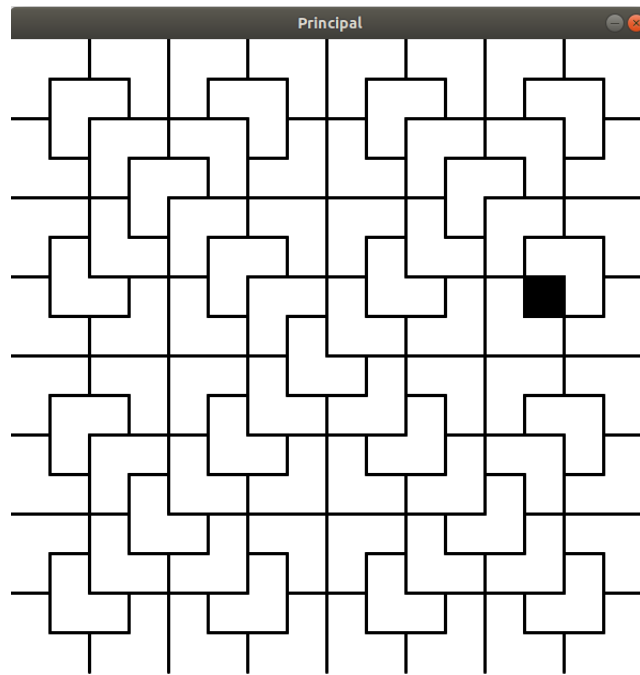
Se debe hacer un programa en Java para resolver el problema. Pueden usar una interfaz gráfica para mostrar el resultado.

El problema se puede ver como colorear un tablero con L's, de tal forma que 3 cuadros con el mismo color son adyacentes formando la L, pero estos no pueden ser adyacentes a otra L con el mismo color.

El siguiente ejemplo muestra una coloración con el *cuadro especial* en (13,6) y $m = 2^4 = 16$:



Aquí el mismo ejemplo pero sin colores:



Esto se hizo usando *core.jar* de Processing, pero lo pueden hacer con cualquier herramienta.

Un color se puede representar como un número o una letra en una matriz. Si no hacen interfaz, deben imprimir la matriz resultante.

La matriz relacionada al ejemplo anterior es la siguiente(en mi implementación):

12	12	13	13	11	11	9	9	12	12	13	13	11	11	9	9
12	2	2	13	11	2	2	9	12	2	2	13	11	2	2	9
14	2	12	12	13	13	2	10	14	2	12	12	13	13	2	10
14	14	12	2	2	13	10	10	14	14	12	2	2	13	10	10
8	8	14	2	12	12	13	13	8	8	14	2	11	11	9	9
8	2	14	14	12	2	2	13	8	2	14	14	11	2	2	9
7	2	2	6	14	2	8	8	7	2	2	6	1	0	2	10
7	7	6	6	14	14	8	2	7	7	6	6	1	1	10	10
12	12	13	13	8	8	7	2	2	6	3	3	11	11	9	9
12	2	2	13	8	2	7	7	6	6	2	3	11	2	2	9
14	2	8	8	7	2	2	6	5	2	2	4	3	3	2	10
14	14	8	2	7	7	6	6	5	5	4	4	2	3	10	10
8	8	7	2	2	6	3	3	8	8	5	2	2	4	3	3
8	2	7	7	6	6	2	3	8	2	5	5	4	4	2	3
7	2	2	6	5	2	2	4	7	2	2	6	5	2	2	4
7	7	6	6	5	5	4	4	7	7	6	6	5	5	4	4

4. Detalles

- Se puede hacer iterativo o recursivo. Son libres de usar cualquier implementación.
- Este programa se entrega de forma individual.
- Si su carpeta contiene un ejecutable(como *.jar) enviarlo como un enlace de dropbox o drive.
- El problema viene explicado en las notas de clase, Sección 3.3.4 pp. 45-46.

5. Entrega

- Deben entregarlo como un archivo comprimido de una carpeta con el mismo nombre.
- La carpeta debe ser: **Programa1_ApellidoPaternoApellidoMaterno**. Por ejemplo **Programa1_PetoGutierrez**.
- Su carpeta debe contener un archivo *readme* que contenga: número de cuenta, nombre completo, correo y las instrucciones para compilar y ejecutar su programa.
- El asunto debe ser: **[AAlgoritmos]ProgramaN**, o sea, **[AAlgoritmos]Programa1** en este caso.