

Análisis de Algoritmos I

Profesora: Luz Gasca Soto

Ayudantes: Bernal Cedillo

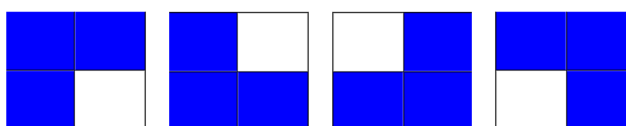
Enrique Antonio y García Flores Jorge Luis

Práctica 1: Adoquinamiento

Agosto, 2017

1 Introducción

El problema del adoquinamiento consiste en utilizar adoquines de la siguiente forma:



Para cubrir completamente la superficie de un área de $m \times m$.

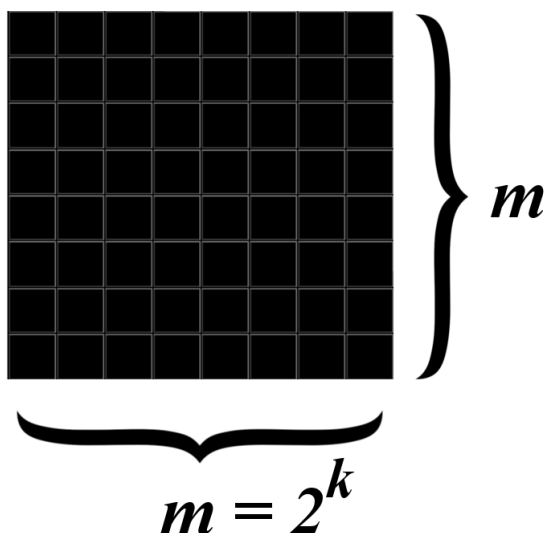
En otras palabras, llenar cada localidad de una matriz de $m \times m$ utilizando únicamente patrones como los anteriormente descritos.

2 Descripción

2.1 Entrada

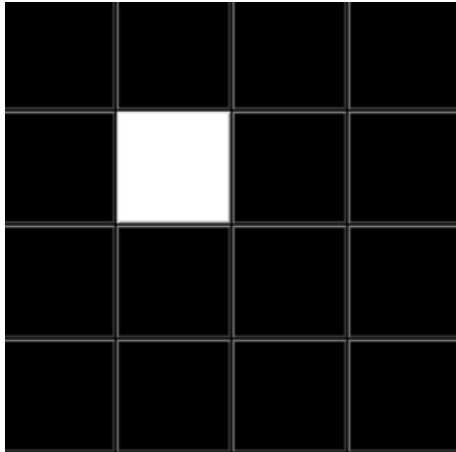
El programa a implementar debe recibir como única entrada un entero k , el cual indica el tamaño de nuestra cuadrícula de la siguiente forma:

$$m = 2^k$$



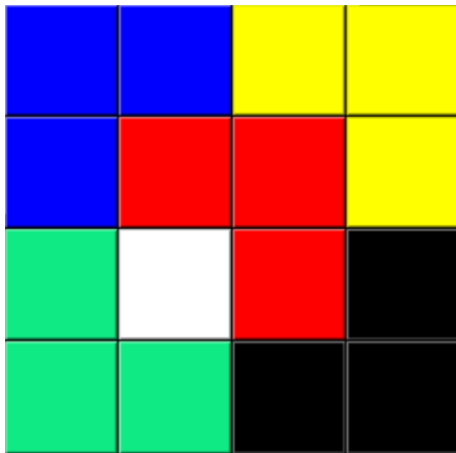
2.2 Adoquín especial

Como parte del problema, se indica que la cuadrícula inicial comienza con un "adoquín especial" que ocupa una sola localidad en la matriz. El programa debe implementarse de manera que en cada ejecución, el adoquín especial aparezca en una posición aleatoria de la cuadrícula. Dicho adoquín debe ser el único de color blanco.



2.3 Salida

Una vez resuelto el problema del adoquín (de manera iterativa o recursiva), debe mostrarse una representación gráfica de la cuadrícula solucionada. Debe ser posible distinguir cada conjunto de 3 adoquines en la solución (los patrones básicos presentados en la Introducción) ya sea utilizando etiquetas, colores, números, o cualquier relación que los agrupe.



3 Bibliotecas de apoyo

3.1 Java (Processing)

Para la implementación en Java, se proporciona un proyecto base en NetBeans que utiliza la biblioteca Processing para el manejo de la interfáz gráfica. Para cualquier duda referirse a la documentación correspondiente:

<https://processing.org/reference/>

3.2 Python (Tkinter)

Si el lenguaje a utilizar se trata de Python, se proporciona un módulo de ejemplo que utiliza la biblioteca Tkinter para dibujar en pantalla la cuadrícula a utilizar.

<http://effbot.org/tkinterbook/>

3.3 Otras

La biblioteca a utilizar para la interfaz grafica es libre de elegirse. Siempre y cuando el programa sea implementado utilizando Java o Python.

4 Fecha de Entrega

La fecha de entrega de la práctica es el día 3 de Septiembre de 2017.

Al correo enrique.bernal@ciencias.unam.mx

Con asunto '[Practica01]ApellidoapaternoPrimernombre'

4.1 Adjuntos

-) Readme.txt, con instrucciones de cómo se recibe el parámetro k , nombre completo del alumno y posibles configuraciones adicionales para las bibliotecas utilizadas.
-) Implementación correspondiente al problema en Java o Python.

Todo lo anterior en un archivo .zip/.rar/.tar.gz con el mismo nombre del asunto del correo.