



Práctica de la  
**Asignatura Programación III**  
Versión 1.0

Escuela Técnica Superior de  
Ingeniería Informática

**UNED**

**Curso 2010-2011**

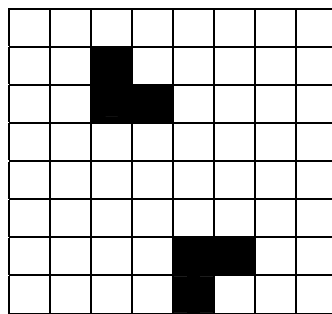


## 1.- ENUNCIADO DE LA PRÁCTICA

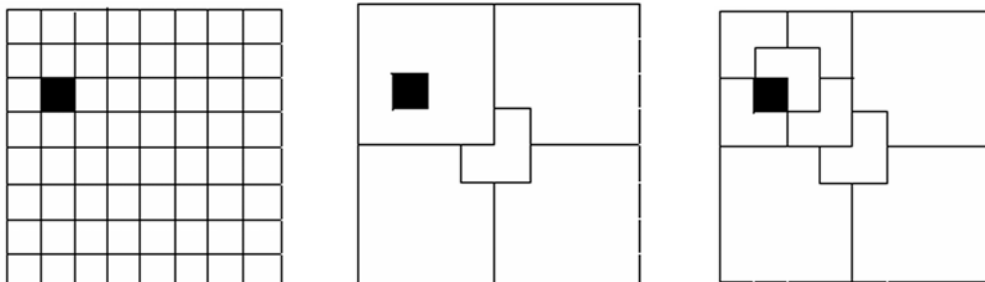
Un trimino en L es una figura formada por 3 cuadrados de dimensión 1x1 agrupados en alguna de las formas siguientes:



Los triminos pueden colocarse en retículas (tablas) de tamaño  $N \times N$  como muestra la figura:



El problema consiste en partir de una cuadrícula con una (y solo una) casilla marcada y rellenarla con triminos de manera que cubran la totalidad de la tabla. Por ejemplo, así:



En general no es posible realizar esta operación salvo que se cumpla que  $N=2^k$ , en cuyo caso el problema siempre tiene solución.

Se pide realizar un algoritmo que proporcione una solución al problema y que permita cubrir el tablero completamente de triminos, salvo una casilla inicial dada conocida con antelación.

## 2.- REALIZACIÓN DE LA PRÁCTICA

La práctica constará de una memoria y de un programa Java original y correctamente compilable que resuelva el problema planteado mediante alguno de los esquemas algorítmicos expuestos en la asignatura.

En la memoria se detallarán las respuestas a las preguntas planteadas en el siguiente apartado y un listado comentado del código fuente. Se valorará que el código esté comentado de forma que se facilite su lectura.

### 2.1.- Diseño del Algoritmo

El programa que desarrolle el algoritmo solución debe seguir un diseño orientado a objetos. La solución deberá motivarse y realizar un análisis teórico de la complejidad del algoritmo resultante.

Se desarrollarán también las respuestas a las siguientes preguntas:

1. Describir el tipo de problema y esquema algorítmico utilizado, que debe ser el de menor coste computacional de los posibles que se pudieran aplicar.
2. Exponer alternativas al esquema utilizado. ¿Hay alguna otra forma de resolverlo? ¿En qué medida es peor que la elegida para la solución? Comentar la decisión tomada.
3. Analiza el coste en términos de tratabilidad computacional.

### 2.2.- Argumentos y Parámetros

La práctica se invoca usando la siguiente sintaxis:

```
trimino [-t][-h] x y [d] [fichero]
```

Los argumentos son los siguientes :

- **-t** : Traza cada paso de manera que se describa la aplicación del algoritmo utilizado
- **-h** Muestra una ayuda y la sintaxis del comando
- **x** e **y** son las coordenadas de la casilla inicial marcada.
- **d** es la dimensión del tablero. Por defecto se entiende 8x8
- **fichero** es el nombre del fichero que se creará y almacenará la salida. Si el fichero ya existe, el comando dará un error. En caso de que no se indique fichero, el programa devolverá la solución por la salida estándar.

Si no tiene argumentos, el programa muestra la matriz inicial, y la matriz final resuelta.

### 2.3.- Funcionalidad de la práctica

El programa se deberá poder invocar desde la línea de comandos de Linux (ó Windows) con la siguiente sintaxis:

```
$>java trimino [-t][-h] x y [d] [fichero]
```

Y devolverá bien en un fichero o bien por la salida estándar, la solución al problema.

El comando se invoca desde la línea de comandos (Ej. windows):

```
C:\>java trimino 4 3 solucion.txt <ENTER>
```

o también hacia la salida estándar (stdout):

```
C:\>java trimino 4 3 > solucion.txt <ENTER>
```

o también por pantalla:

```
C:\>java trimino 4 2 <ENTER>
```

La salida se representa mediante caracteres diferentes por cada trimino. Se muestra a continuación el resultado de los primeros pasos del desarrollo de la solución.

```
1 1 2 2 . . . .
1 3 3 2 . . . .
4 # 3 5 . . . .
4 4 5 5 6 . . . .
. . . 6 6 . . . .
. . . . . . . .
. . . . . . . .
. . . . . . . .
```

Se podrá usar para rellenar los triminos del juego habitual de caracteres alfanuméricos los números del 0 al 9 y las letras de la 'a' a la 'z' (y sus mayúsculas). En caso de que falten, se hará un recorrido circular sobre ellos. Se asume que es posible, aunque poco probable, que dos triminos adyacentes compartan el mismo carácter. También podrán usarse representaciones gráficas, pero no otro juego alternativo de caracteres alfanuméricos. El carácter '#' se reserva para la casilla inicial marcada.

Todos estos requisitos son obligatorios para aprobar la práctica.

### 3.- IMPLEMENTACIÓN

#### 3.1.- Compiladores

##### Compilador de Java

Se usará Java 1.4 o superior. Todo lo necesario está disponible en <http://java.sun.com>. Cualquier entorno de desarrollo es útil para realizar la práctica. Basta en cualquier caso un editor de línea aunque se recomienda BlueJ (<http://www.bluej.org>)

#### 3.2.- Estudio del Coste

Estudiar y calcular la complejidad del algoritmo. Realizar un estudio teórico del coste del algoritmo implementado.

### 4.- ENTREGA DE LA PRÁCTICA

1. La asistencia a las sesiones de prácticas es obligatoria **SIN** excepciones.
2. El calendario y procedimiento para asistir a las sesiones de prácticas **está publicado en su Centro Asociado**.
3. El plazo de entrega de la documentación y de la práctica **lo establece el Tutor de prácticas de cada Centro Asociado**.
4. El Tutor califica la práctica, informa al alumno y en su caso la revisa de acuerdo con los horarios y procedimiento que establezca el Centro Asociado.
5. La Sede Central NO informa de las calificaciones de las prácticas, lo hace el Tutor o el Centro Asociado.
6. TODOS los alumnos deberán registrarse a través del Curso Virtual (acceso desde CiberUNED) con el Tutor/a con el que hayan asistido a las sesiones presenciales obligatorias a fin de que su práctica pueda ser calificada.
7. No se recogerá documentación alguna de alumnos que no hayan asistido a las sesiones obligatorias.
8. La práctica se debe aprobar en la misma o anterior convocatoria para que se pueda calificar la asignatura. En caso contrario la calificación será de suspenso.
9. La práctica se entregará empaquetada en un \*.jar tanto en el entorno WebCT como al Tutor. La falta de cualquiera de ellas será motivo suficiente para quedar excluida de la convocatoria.

## TUTORES

1. La práctica se califica mediante un entorno web accesible desde los Cursos Virtuales.
2. Los tutores se podrán dar de alta en la aplicación de entrega de notas de la práctica, sin necesidad de ponerse en contacto con el equipo docente. En el Foro de tutores recibirán las instrucciones pertinentes.
3. Los alumnos se dan de alta en la aplicación de corrección y calificación de prácticas accesible desde WebCT de acuerdo con las indicaciones del Tutor. Aquel alumno que no se dé de alta no aparecerá en la lista de presentados a la misma y no constará su nota en el entorno de calificación y revisiones.

### Material que hay que entregar al Tutor

Se confeccionará una memoria escrita con índice:

1. Respuesta a las cuestiones planteadas en este enunciado.
2. Un ejemplo de ejecución para distintos tamaños del problema: 8, 16, 32, 64 etc.
3. Estudio del coste del algoritmo.
4. Un listado del código fuente COMPLETO.

Se adjuntará **un CD** con el nombre y apellidos del alumno escrito en el mismo y conteniendo:

1. Fuentes de cada uno de los módulos.
2. Ejecutable \*.jar del programa principal.
3. Ficheros de los casos de prueba.
4. Memoria de la práctica

El disco y toda la documentación que se entregue deberá tener bien legible: **Nombre y apellidos, C. Asociado, teléfono y e-mail.**

El alumno debe asegurarse de que no se da ninguna de las siguientes circunstancias, ya que implican automáticamente una calificación de SUSPENSO:

- **Código:** El código no compila, no está desarrollado en Java, no se corresponde con el pseudocódigo recogido en la documentación, no es original, está copiado de la red, academia, compañero, etc., o no sigue un diseño OO encapsulado o modular.
- **Ejecutable:** El ejecutable no termina, se queda sin memoria con ejemplares pequeños o aborta sin justificación. El ejecutable no lee los ficheros previstos en el formato adecuado. No trata los argumentos o no se ajusta a las especificaciones.

- **Documentación:** No se presenta en papel o está incompleta.
- **Soporte:** No se puede leer, o contiene un virus de cualquier tipo. A este respecto, las prácticas en las que se detecte cualquier tipo de virus estarán suspensas.

\*\*\*\*

**La práctica debe realizarse de forma individual y se entregará al TUTOR en las fechas que fije cada Centro Asociado.**

\*\*\*\*

El Centro Asociado publicará las notas de las prácticas. El equipo docente **NO INFORMARÁ DE LAS NOTAS DE PRÁCTICAS**, debiendo el Alumno dirigirse al Centro Asociado para conocerlas.



## **ANEXO I**

### **Bibliografía y tutoriales de Java**

La referencia más útil es el libro “*Piensa en Java*” (“*Think in Java*”) de Bruce Eckel, disponible en varias editoriales. Hay una versión completa y gratuita en la red en <http://www.mindview.net/Books/TIJ>.

En <http://www.javahispano.org/> hay multitud de tutoriales en español.

En la web de BlueJ también hay mucha información interesante.

### **Alumnos sin TUTOR**

No hay sesiones de prácticas para alumnos sin Tutor, ya que no es posible según la normativa de la UNED matricularse en aquellos Centros que no disponen de Tutor de esta asignatura, salvo los **alumnos estudiando en el EXTRANJERO**, que se deberán poner en contacto con el profesor Antonio Sarasa en el FORO de “alumnos fuera de España” de WebCT.

El resto de las dudas las resolverá el TAR, o en su defecto el equipo docente a través de los foros de WebCT.