

# Trabajo Práctico 2

## Lectura de mapas

### Introducción a la Computación

1<sup>er</sup> cuatrimestre 2019

#### Observaciones generales:

- El trabajo se debe realizar en grupos de tres personas.
- El código fuente debe estar bien comentado. Cualquier aclaración adicional que se considere necesaria debe ser incluida como comentarios.
- El código fuente debe enviarse por mail a la lista de docentes de la materia: `icm-doc@dc.uba.ar`, indicando los integrantes del grupo.
- El programa entregado debe compilar correctamente usando Python 3.6.7, que está instalado en las computadoras del laboratorio.
- Se evaluará la correctitud, claridad y modularidad del código entregado.
- La fecha límite de entrega es el jueves 23 de mayo a las 23:59.

Este trabajo se basa en interpretar un archivo de texto plano como una grilla que determina un mapa con dos tipos de terreno: *espacios vacíos* o *paredes sólidas*. Para comenzar, dado un archivo, se debe verificar que su contenido sea válido para interpretarse como mapa:

- Cada línea del archivo de texto corresponde a una fila de la grilla.
- Cada fila de la grilla está formada únicamente por los caracteres 0 (que representa los *espacios vacíos*) o 1 (las *paredes sólidas*).
- Las filas y columnas de una grilla deben formar un rectángulo.

Se pide implementar en Python las siguientes funciones para inputs válidos:

- Contar la cantidad de *paredes* en un mapa.
- Contar la cantidad de *espacios vacíos* que están completamente rodeados de paredes. Como ejemplo, el 0 está completamente rodeado en el siguiente ejemplo:

```
111
101
111
```

- Determinar las dimensiones del mapa: alto (número de filas) y ancho (número de columnas).
- La longitud del “corredor horizontal más largo”, definida como la secuencia horizontal más larga de *espacios vacíos*. Por ejemplo, si una fila del archivo es 011000111100, esa línea tiene un corredor

horizontal de longitud 3. Si ninguna otra fila tiene un corredor horizontal de longitud mayor a 3, entonces ése es el corredor horizontal más largo del mapa. Esta operación debe realizarse en  $O(n)$ , donde  $n$  es la cantidad total de posiciones en el mapa; es decir, la suma de los *espacios vacíos* y las *paredes sólidas*.

- Determinar la *densidad arquitectónica* del mapa, definida como  $\#paredes/\#espacios$  total.

El programa presentado deberá poder ejecutarse desde la línea de comandos y recibirá como argumentos el nombre de la función que se desea ejecutar (por ejemplo, “es\_mapa”) más los argumentos necesarios para dicha función. A continuación se exponen usos del programa con el resultado esperado para el llamado de distintas funciones (sobre archivos con nombres descriptivos de sus contenidos), suponiendo que el archivo Python se llama `tp2.py`. En estos ejemplos, “I” es el comando a ejecutar por línea de comandos; “O” es el resultado impreso por pantalla:

- I: `python tp2.py es_mapa NoRectangulo.txt`  
O: El archivo no representa un mapa: no es rectangular o tiene caracteres distintos de 0 y 1
- I: `python tp2.py es_mapa RectanguloDeLetras.txt`  
O: El archivo no representa un mapa: no es rectangular o tiene caracteres distintos de 0 y 1
- I: `python tp2.py es_mapa BuenMapa.txt`  
O: Sí, el archivo representa un mapa.
- I: `python tp2.py cantidad_paredes MapaCon31Paredes.txt`  
O: El mapa tiene 31 paredes.
- I: `python tp2.py espacios_rodeados MapaCon2Rodeados.txt`  
O: El mapa tiene 2 huecos rodeados de paredes.
- I: `python tp2.py dimensiones Mapa3Filas6Columnas.txt`  
O: El mapa tiene 6 cuadrados de ancho y 3 de alto.
- I: `python tp2.py corredor_horizontal_mas_largo 1FilaDeCeros7Columnas.txt`  
O: El corredor horizontal más largo tiene longitud 7.
- I: `python tp2.py densidad MitadYMitad.txt`  
O: La densidad arquitectónica es de 0.5.

#### Observaciones:

- Definir todas las funciones auxiliares que se consideren necesarias.
- Si se llama una función sobre un archivo no válido, en vez del funcionamiento usual se debe imprimir en pantalla que el archivo no representa un mapa.
- No está permitida la importación de ninguna biblioteca, con la excepción de `sys` para el manejo de los argumentos del programa principal con `sys.argv`.