

Laboratorio 3

Todas las coordinaciones

FACULTAD DE INGENIERIA
Departamento de Ingeniería Informática
Análisis de algoritmos y estructuras de datos



26/04 /2017

Los laberintos son lugares compuestos básicamente por paredes y pasillos dispuestos de tal manera que confundan a las personas que se encuentren en él, dificultando así que se pueda encontrar una salida. Estos diseños existen desde la antigüedad (figura 1), siendo los más antiguos los laberintos de forma cuadrada y rectangular.



Figura 1: Laberinto indio de hace 2000 años¹.

Se solicita que a partir de la representación de un laberinto (figura 2) logre encontrar el camino más corto para salir e indique la ruta que se debe seguir para encontrar una llave y luego salir del laberinto.

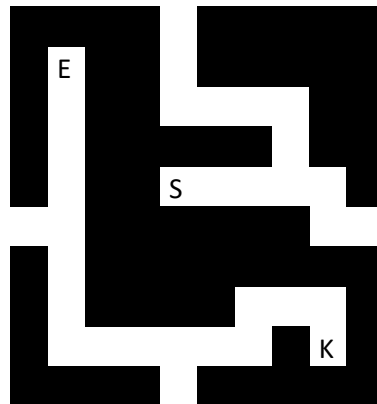


Figura 2: Laberinto.

[1] <http://www.ancient-origins.es/noticias-historia-arqueologia/laberinto-hace-2000-%C3%B1os-descubierto-la-india-sigue-el-mismo-patr%C3%B3n-dise%C3%B1o-que-uno-griego-1200-c-002792>

Laboratorio 3

Todas las coordinaciones

FACULTAD DE INGENIERIA
Departamento de Ingeniería Informática
Análisis de algoritmos y estructuras de datos



26/04 /2017

El laberinto será representado por los puntos de inicio y fin de los caminos que hay dentro de él, los cuales pueden estar o no unidos a otros caminos (figura 3).

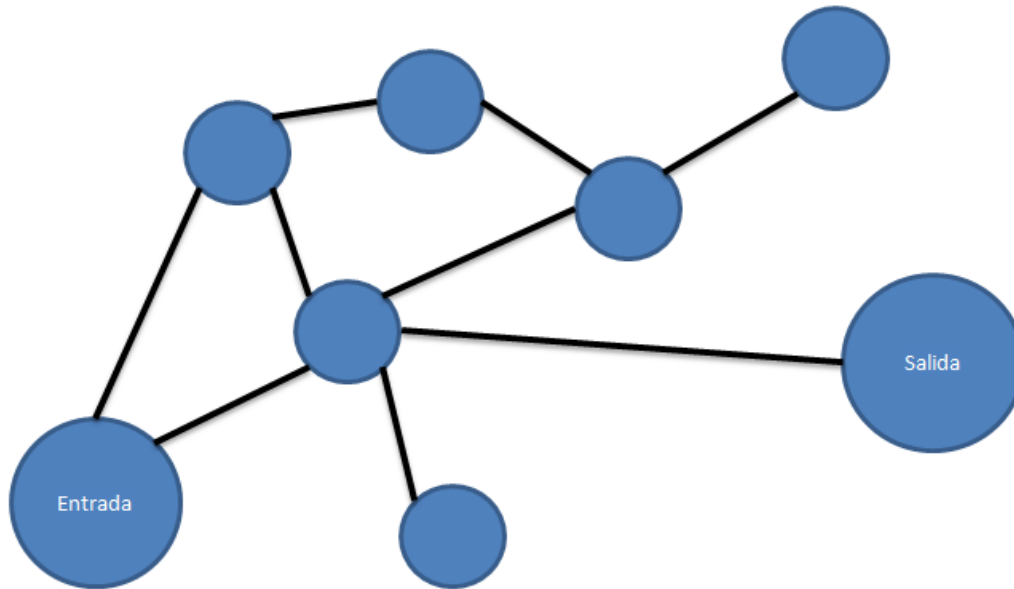


Figura 3: Representación de Laberinto.

Para esto, el programa debe recibir un archivo de texto llamado Entrada.in (figura 3). El que contendrá en la primera línea la cantidad de puntos de unión que hay en el laberinto (inicio y final de caminos), luego en la segunda, tercera y cuarta línea se indica en qué punto se encuentra la entrada, llave y salida del laberinto respectivamente. Finalmente, las siguientes indican cómo están conectados los caminos indicando su punto de inicio, fin y la longitud del camino de la forma: <punto inicio> <punto fin> <longitud>. La entrada será representada por la letra “E”, la salida por la letra “S” y la letra “K” representa la llave que se debe encontrar previamente para abrir la salida.

Laboratorio 3

Todas las coordinaciones

FACULTAD DE INGENIERIA
Departamento de Ingeniería Informática
Análisis de algoritmos y estructuras de datos



26/04 /2017

16		
E 3		
K 14		
S 9		
1	2	10
15	16	5
4	5	1
10	12	2
3	7	1
12	15	12
5	8	3
8	10	4
4	9	5
11	14	1
9	14	22
2	3	2
14	16	8
2	6	3
14	13	1
3	10	3
11	12	5

Figura 3: Archivo de entrada.

El archivo de entrada mostrado en la figura 3 representa el laberinto mostrado en la figura 4 y la solución de este es mostrada en las figuras 6 y 7:

Laboratorio 3

Todas las coordinaciones

FACULTAD DE INGENIERIA
Departamento de Ingeniería Informática
Análisis de algoritmos y estructuras de datos



26/04 /2017

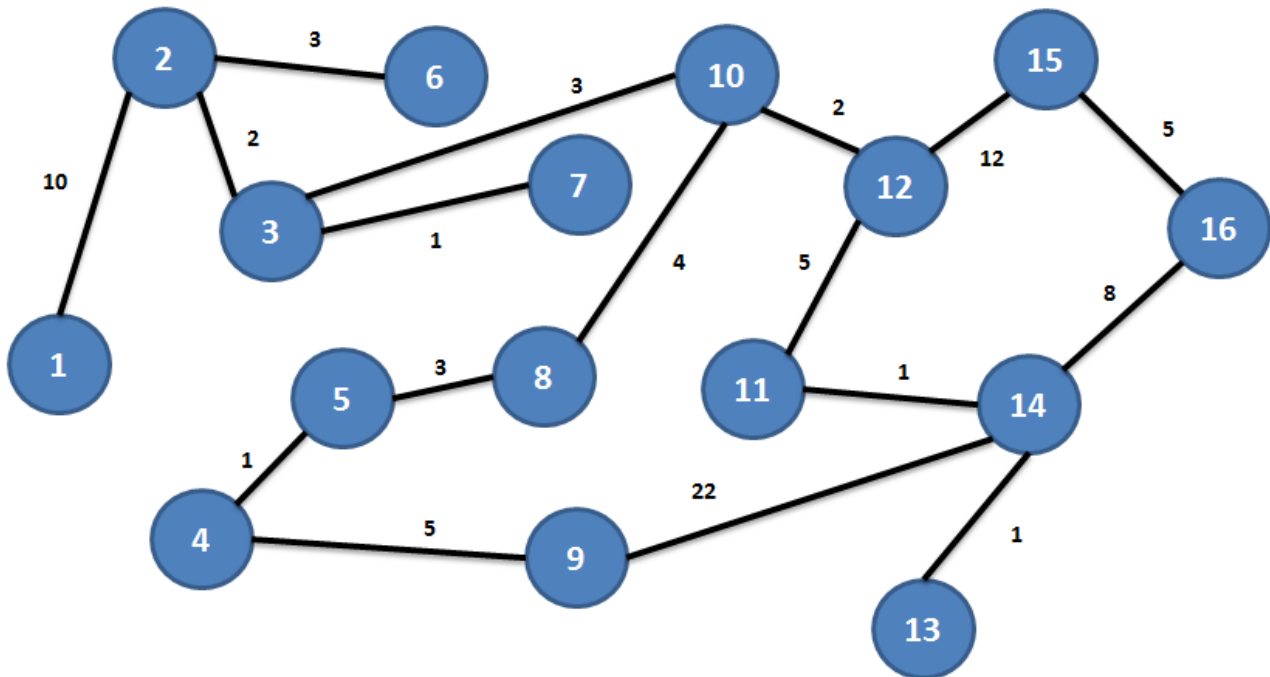


Figura 5: Laberinto representado.

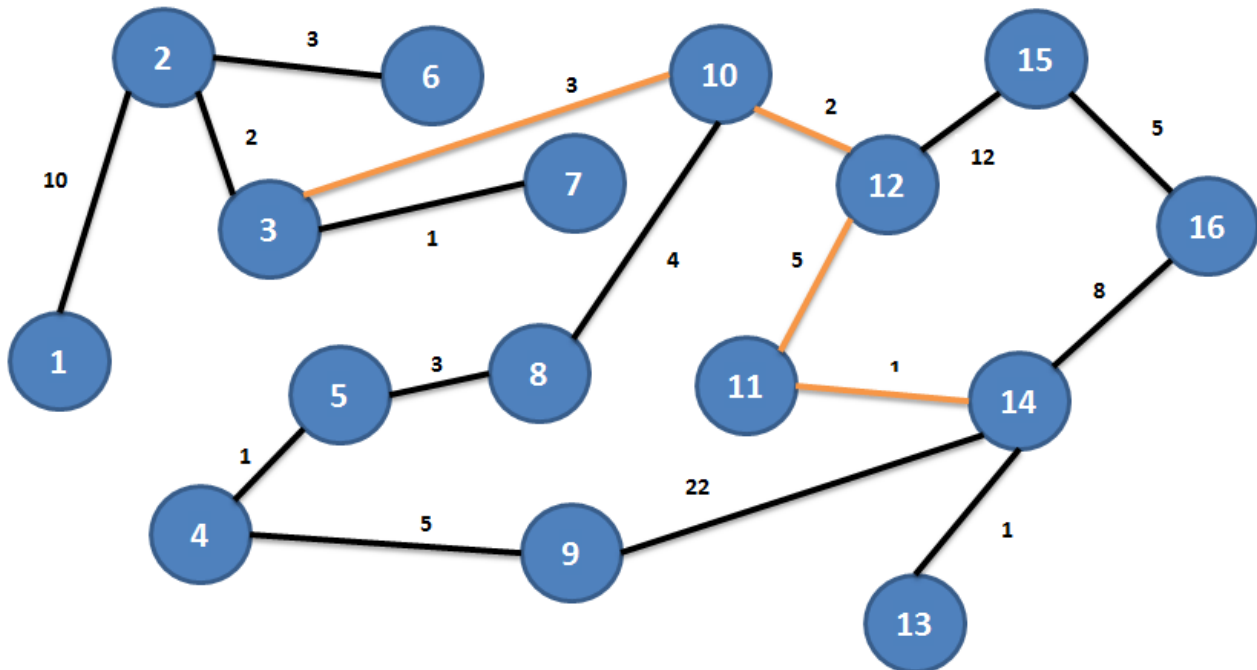


Figura 6: Solución para encontrar llave de la salida del laberinto representado.

Laboratorio 3

Todas las coordinaciones

FACULTAD DE INGENIERIA
Departamento de Ingeniería Informática
Análisis de algoritmos y estructuras de datos



26/04 /2017

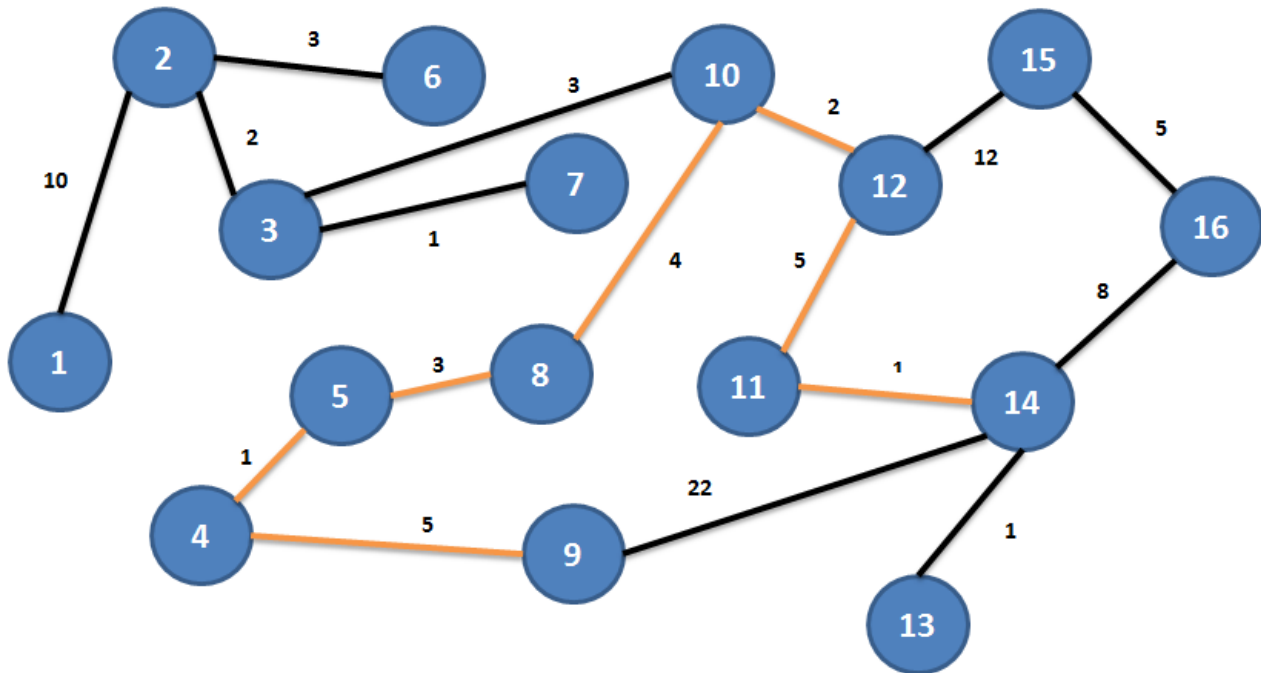


Figura 7: Solución para encontrar la salida del laberinto representado desde la llave.

Como resultado debe generar un segundo archivo de texto llamado Salida.out (figura 8), que debe contener dos secuencias de puntos por lo que es necesario pasar para salir del laberinto. La primera debe ser la secuencia para ir desde la entrada hasta la llave y la segunda la secuencia que permite ir hasta la salida una vez se a recuperado la llave. Junto a cada secuencia se debe indicar la distancia total a recorrer.

Laboratorio 3

Todas las coordinaciones

FACULTAD DE INGENIERIA
Departamento de Ingeniería Informática
Análisis de algoritmos y estructuras de datos



26/04 /2017

```
Salida.out: Bloc de notas
Archivo  Edición  Formato  Ver  Ayuda

Camino entrada a llave:
3 - 10 - 12 - 11 - 14
Distancia recorrida: 11

Camino llave a salida:
14 - 11 - 12 - 10 - 8 - 5 - 4 - 9
Distancia recorrida: 21
```

Figura 8 Archivo de salida.

Fecha de entrega: 18/05/2017 hasta las 23:30 hrs.

Instrucciones de entrega:

- Archivo PDF con el informe (incluye manual de usuario).
- Se debe calcular el $T(n)$ y el O .
- Código fuente en archivos .c y .h (no entregar proyectos de ninguna IDE).
- El código debe permitir ser compilado en ambiente Windows y Linux por lo que se sugiere usar ANSI C.