

**ESTRUCTURA DE DATOS Y ANÁLISIS DE ALGORITMOS  
MANUAL DE USUARIO**

**CRISTIAN EDUARDO ESPINOZA SILVA**

Profesor: Alejandro Cisterna  
Ayudante: -  
Fecha de Entrega: 6 de abril de 2017

Santiago de Chile

1 - 2017

## TABLA DE CONTENIDO

CAPÍTULO 1. Introducción .....	1
CAPÍTULO 2. Instrucciones .....	2
2.1 Descarga de mingw .....	2
2.2 Compilación .....	2
2.2.1 Windows: .....	2
2.2.2 Linux .....	5
2.3 Ejecución .....	8
2.3.1 Windows.....	8
2.3.2 Linux .....	9
CAPÍTULO 3. Funcionalidades .....	10
CAPÍTULO 4. Posibles errores .....	14

## ÍNDICE DE FIGURAS

Figura 1: Ventana ejecutar en sistema operativo Windows. ....	2
Figura 2: Consola de Windows. ....	3
Figura 3: Crear carpeta y almacenar software dentro.....	3
Figura 4: Buscando ruta de la carpeta creada.....	4
Figura 5: Compilando software.....	4
Figura 6: Creado el ejecutable del software. ....	5
Figura 7: Abrir consola de Linux o Terminal. ....	5
Figura 8: Ventana de la consola de Linux o Terminal. ....	6
Figura 9: Carpeta con el software dentro. ....	6
Figura 10: Consola de Linux con la ruta de la carpeta que contiene el software..	7
Figura 11: Compilación de software. ....	7
Figura 12: Carpeta con el archivo ejecutable .....	8
Figura 13: Software ejecutado.....	9
Figura 14: Software ejecutado.....	9
Figura 15: Menú principal del software. ....	10
Figura 16: Cargando el archivo al software. ....	11
Figura 17: Mostrando el laberinto por pantalla. ....	11
Figura 18: Buscando camino hacia la Llave y luego hacia la salida. ....	12
Figura 19: Cerrando el software.....	12
Figura 20: Carpeta con el archivo de “salida.out” del software .....	13
Figura 21: Salida final del software. ....	13
Figura 22: Posible error dentro de software. ....	14

## **CAPÍTULO 1. INTRODUCCIÓN**

En el siguiente documento se presenta una forma adecuada, explicativa y simple de cómo utilizar el software hecho.

El software principalmente consiste en la resolución del problema entregado en el laboratorio 1, que el objetivo principal es encontrar el camino correcto hacia la llave y luego hacia la salida del “Laberinto”.

Las reglas del “Laberinto” son diferentes a las que estamos acostumbrados, ya que se podría decir que es un “Laberinto circular”, de manera que si uno sale por un lugar inferior procede a entrar por la parte superior, así mismo si uno procede a salir por los costados entrara por el costado opuesto.

Dentro del transcurso del manual se procede a explicar desde el primer paso que se debe hacer para poder utilizar el software de una manera correcta.

Se recomienda tener poseer un lenguaje técnico mayor al promedio para poder aprovechar las ventajas que entrega la aplicación.

Por ultimo debemos destacar que el software esta implementado en un paradigma imperativo, que principalmente consta de declaraciones que se ejecutan secuencialmente.

## CAPÍTULO 2. INSTRUCCIONES

A continuación, se procede a explicar cada uno de los pasos a seguir para poder compilar el software que soluciona el “Laberinto”.

### 2.1 DESCARGA DE MINGW

Para poder compilar el software credo se necesita descargar lo siguiente, con el fin que los comandos ingresados sean conocidos por su equipo.

Se puede obtener del siguiente link:

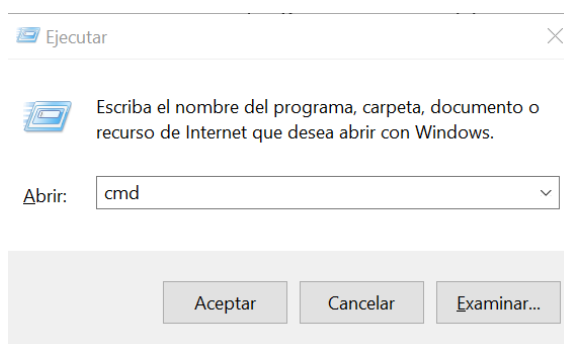
- <https://www.fdi.ucm.es/profesor/luis/fp/devtools/MinGW.html>

### 2.2 COMPILACIÓN

#### 2.2.1 Windows:

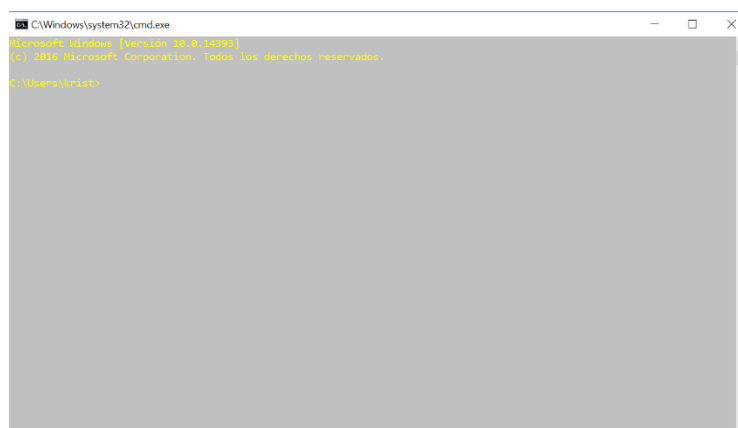
1. Debemos abrir la consola de Windows, existen dos opciones las cuales son:
  - a. Apretar Windows + R
  - b. Abrir la ventana de ejecutar y escribir “cmd”.

Una de las opciones se puede apreciar en la Figura 1, que viene a continuación:



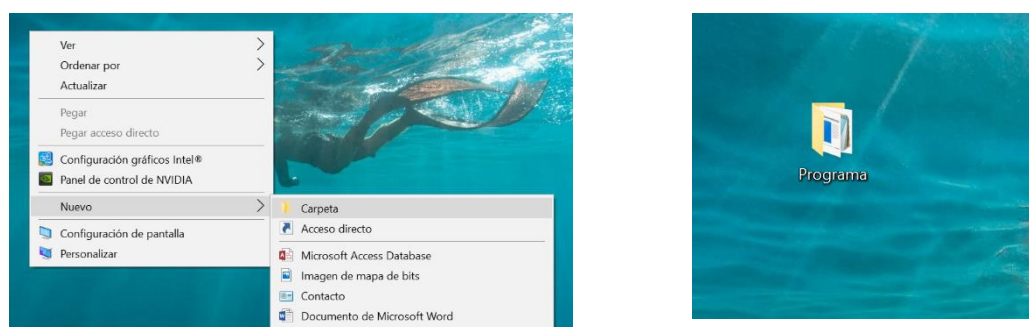
*Figura 1: Ventana ejecutar en sistema operativo Windows.*

2. Luego de realizar el anterior paso, se abrirá la consola como mostrará en la Figura 2:



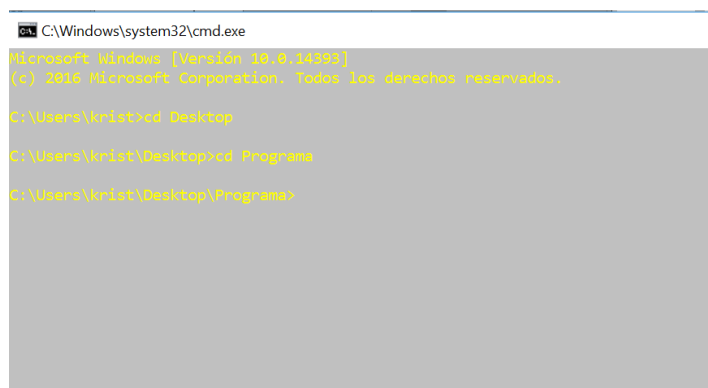
*Figura 2: Consola de Windows.*

3. Almacenar el software dentro de una carpeta para poder tener todo lo necesario dentro de ella, lo recomendado es crear dicha carpeta en el escritorio de su máquina, como se muestra en la Figura 3:



*Figura 3: Crear carpeta y almacenar software dentro.*

4. Luego volvemos a la consola de Windows, para poder ejecutar el software y seguir los siguientes pasos:
  - a. Primer paso, dejamos la consola de Windows en la ruta donde tenemos almacenado nuestro programa, de la manera que muestra en la Figura 4:



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 10.0.14393]
(c) 2016 Microsoft Corporation. Todos los derechos reservados.

C:\Users\kr1st>cd Desktop

C:\Users\kr1st\Desktop>cd Programa


C:\Users\kr1st\Desktop\Programa>
```

*Figura 4: Buscando ruta de la carpeta creada.*

b. Luego comenzamos la compilación, escribiendo lo siguiente en la consola de Windows:

- gcc (Nombre del archivo).c -o (Nombre del ejecutable)
- (Nombre del ejecutable).exe

La Figura 5 entrega las instrucciones con mayor claridad:



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 10.0.14393]
(c) 2016 Microsoft Corporation. Todos los derechos reservados.

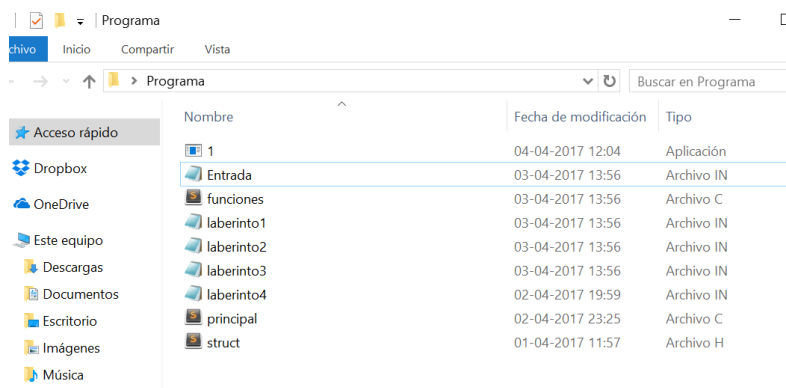
C:\Users\kr1st>cd Desktop

C:\Users\kr1st\Desktop>cd Programa

C:\Users\kr1st\Desktop\Programa>gcc principal.c -o 1_
```

*Figura 5: Compilando software*

- c. Luego se crea el archivo ejecutable en la carpeta donde tenemos almacenado el software, como muestra la Figura 6:

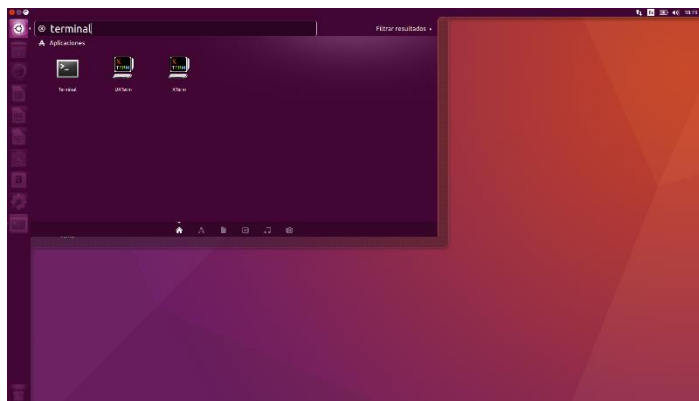


*Figura 6: Creado el ejecutable del software.*

### 2.2.2 Linux

1. Debemos abrir la consola de Linux o también llamada terminal, existen dos opciones las cuales son:
  - a. Apretar clic derecho y colocar la opción abrir terminal.
  - b. Buscar en los programas de Linux y colocar “Terminal”

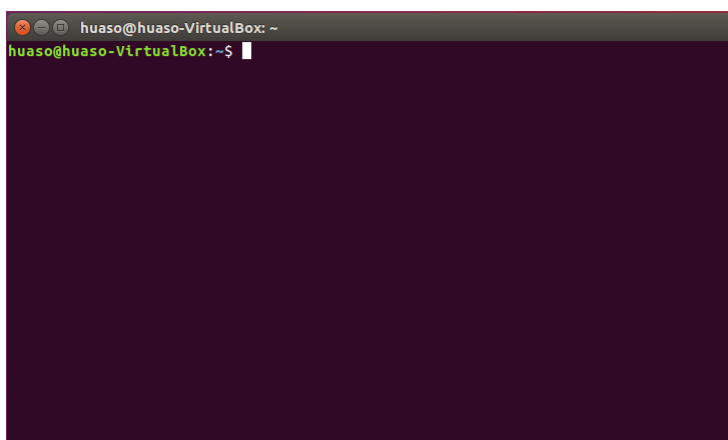
Una de las opciones se puede apreciar en la Figura 7 que se observa abajo:



*Figura 7: Abrir consola de Linux o Terminal.*

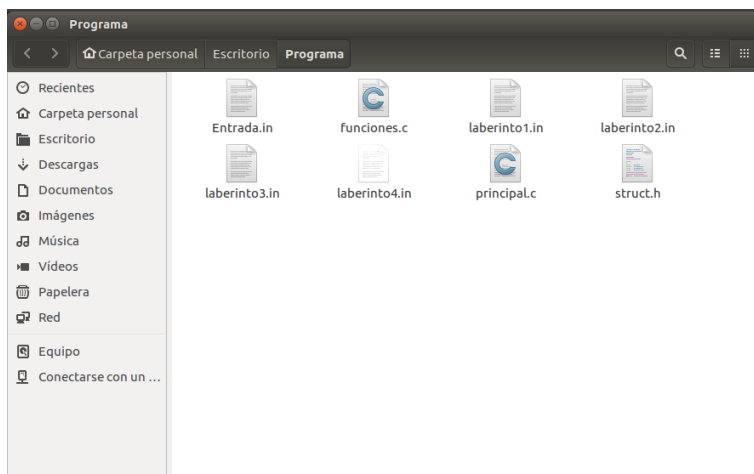


2. Luego de realizar el anterior paso, se abrirá la consola como mostrará la Figura 8:



*Figura 8: Ventana de la consola de Linux o Terminal.*

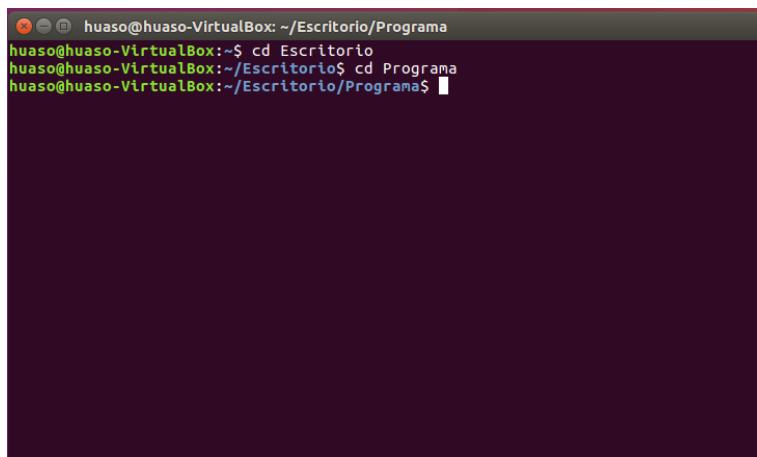
3. Almacenar el software dentro de una carpeta para poder tener todo lo necesario dentro de ella, lo recomendado es crear dicha carpeta en el escritorio de su máquina, como se muestra en la Figura 9:



*Figura 9: Carpeta con el software dentro.*

4. Luego volvemos a la consola de Linux, para poder ejecutar el software como muestra en la Figura 10:

- a. Primer paso, dejamos la consola de Linux en la ruta donde tenemos almacenado nuestro programa, de la manera que muestra la siguiente imagen:

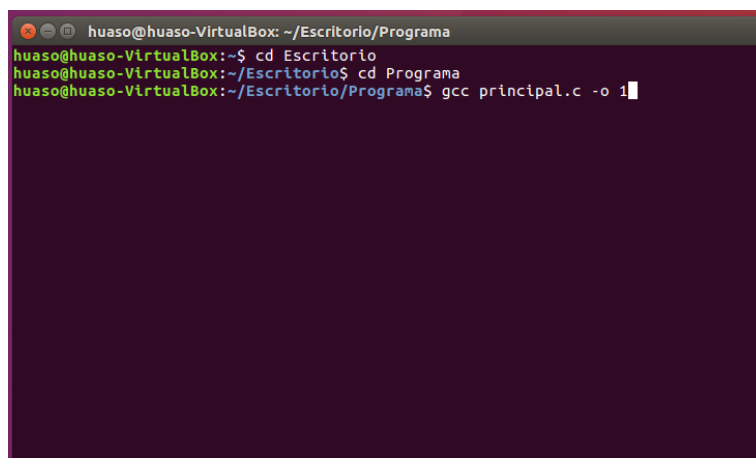


```
huaso@huaso-VirtualBox: ~/Escritorio/Programa
huaso@huaso-VirtualBox:~$ cd Escritorio
huaso@huaso-VirtualBox:~/Escritorio$ cd Programa
huaso@huaso-VirtualBox:~/Escritorio/Programa$
```

*Figura 10: Consola de Linux con la ruta de la carpeta que contiene el software.*

- b. Luego comenzamos la compilación, escribiendo lo siguiente en la consola de Linux:
- gcc (Nombre del archivo).c -o (Nombre del ejecutable)
  - ./(Nombre del ejecutable)

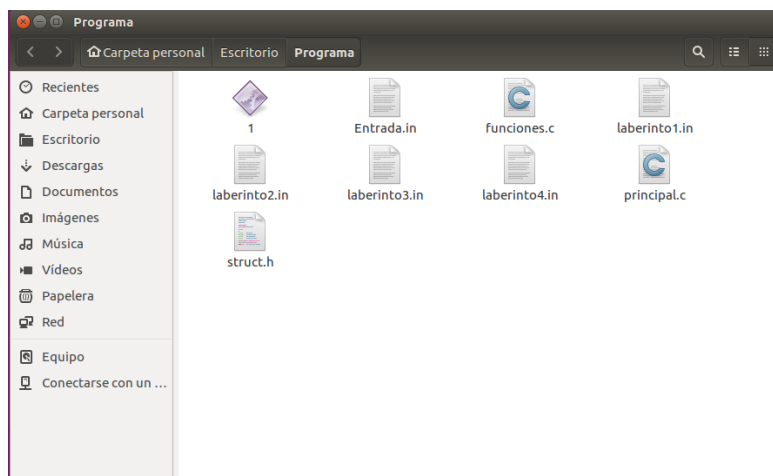
La Figura 11 entrega las instrucciones con mayor claridad:



```
huaso@huaso-VirtualBox: ~/Escritorio/Programa
huaso@huaso-VirtualBox:~$ cd Escritorio
huaso@huaso-VirtualBox:~/Escritorio$ cd Programa
huaso@huaso-VirtualBox:~/Escritorio/Programa$ gcc principal.c -o 1
```

*Figura 11: Compilación de software.*

- c. Luego se crea el archivo ejecutable en la carpeta donde tenemos almacenado el software, como muestra Figura 12:



*Figura 12: Carpeta con el archivo ejecutable*

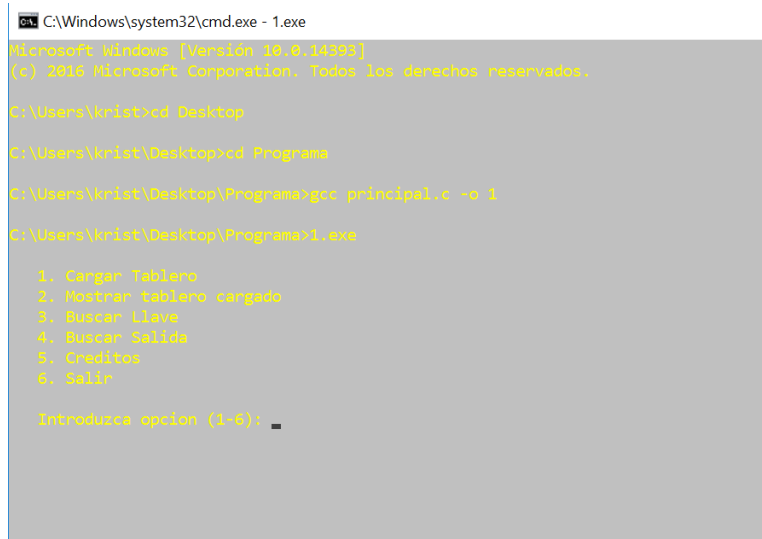
## 2.3 EJECUCIÓN

Teniendo el archivo “.exe” o “.out” en Windows y Linux respectivamente se procede a ejecutarlo cada uno en su ambiente.

Antes de llevar a cabo dicha acción se debe verificar que los caminos de los “Laberinto” estén en la misma carpeta donde se tiene almacenado dicho software, además de corroborar que los archivos donde se tiene almacenado cada uno de los caminos estén con la extensión.in.

### 2.3.1 Windows

1. Se vuelve a la consola de comandos de Windows, colocando la siguiente línea para efectuar la ejecución, en la Figura 13 se puede apreciar de mejor forma:



```

C:\Windows\system32\cmd.exe - 1.exe
Microsoft Windows [Versión 10.0.14393]
(c) 2016 Microsoft Corporation. Todos los derechos reservados.

C:\Users\krist>cd Desktop

C:\Users\krist\Desktop>cd Programa

C:\Users\krist\Desktop\Programa>gcc principal.c -o 1

C:\Users\krist\Desktop\Programa>1.exe

1. Cargar Tablero
2. Mostrar Tablero cargado
3. Buscar Llave
4. Buscar Salida
5. Creditos
6. Salir

Introduzca opcion (1-6): █

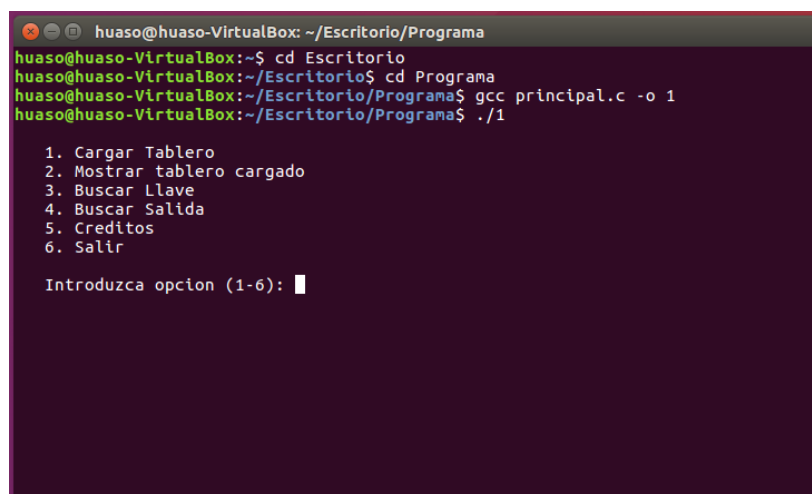
```

*Figura 13: Software ejecutado.*

- Al terminar el proceso anterior, se puede observar en la imagen anterior que el software está ejecutado y listo para ser manipulado.

### 2.3.2 Linux

1. Se vuelve a la consola de comandos de Linux, colocando la siguiente línea para efectuar la ejecución, en la Figura 14 se puede apreciar de mejor forma:



```

huaso@huaso-VirtualBox: ~/Escritorio/Programa
huaso@huaso-VirtualBox:~$ cd Escritorio
huaso@huaso-VirtualBox:~/Escritorio$ cd Programa
huaso@huaso-VirtualBox:~/Escritorio/Programa$ gcc principal.c -o 1
huaso@huaso-VirtualBox:~/Escritorio/Programa$ ./1

1. Cargar Tablero
2. Mostrar tablero cargado
3. Buscar Llave
4. Buscar Salida
5. Creditos
6. Salir

Introduzca opcion (1-6): █

```

*Figura 14: Software ejecutado.*

- Al terminar el proceso anterior, se puede observar en la imagen anterior que el software está ejecutado y listo para ser manipulado.

## CAPÍTULO 3. FUNCIONALIDADES

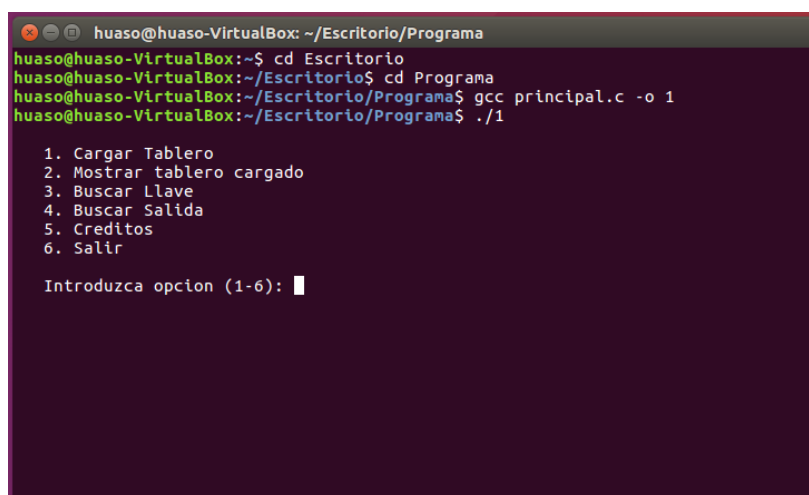
El software analiza el camino posible que se tiene para resolver el “Laberinto”.

El usuario tiene interacción con el software que recae en ingresar el nombre del “Laberinto” que se desea abrir, tener en cuenta que este archivo tendrá una entrada “.in”.

El resultado entregado por el software se verá expuesto en un archivo de salida con extensión “.out”.

A continuación, se realizará un procedimiento de los pasos a seguir para hacer uso del software de una buena manera:

- Primero debemos volver a la consola y ejecutar el software, entregando una ventana como muestra la Figura 15:



```
huaso@huaso-VirtualBox: ~/Escritorio/Programa
huaso@huaso-VirtualBox:~$ cd Escritorio
huaso@huaso-VirtualBox:~/Escritorio$ cd Programa
huaso@huaso-VirtualBox:~/Escritorio/Programa$ gcc principal.c -o 1
huaso@huaso-VirtualBox:~/Escritorio/Programa$ ./1

1. Cargar Tablero
2. Mostrar tablero cargado
3. Buscar Llave
4. Buscar Salida
5. Creditos
6. Salir

Introduzca opcion (1-6): █
```

*Figura 15: Menú principal del software.*

- Luego de eso debemos seleccionar la opción de ingresar el nombre del tablero, como se muestra en la Figura 16:

```

huaso@huaso-VirtualBox: ~/Escritorio/Programa
huaso@huaso-VirtualBox:~/Escritorio/Programa$ gcc principal.c -o 1
huaso@huaso-VirtualBox:~/Escritorio/Programa$ ./1

1. Cargar Tablero
2. Mostrar tablero cargado
3. Buscar Llave
4. Buscar Salida
5. Creditos
6. Salir

Introduzca opcion (1-6): 1
Ingrese el nombre del archivo del laberinto Entrada.in

```

*Figura 16: Cargando el archivo al software.*

- Luego de eso debemos volver a ingresar una opción, en este caso ocuparemos la opción número 2 que nos muestra el tablero leído, se debe hacer como se muestra en la Figura 17:

```

1. Cargar Tablero
2. Mostrar tablero cargado
3. Buscar Llave
4. Buscar Salida
5. Creditos
6. Salir

Introduzca opcion (1-6): 2
*,**,*
*,*S...*K*
*,*****
*,*****
*,...E**
*,*****
*,.....
*,*****
*,*****
*,*****
*,*****
*,*****

1. Cargar Tablero
2. Mostrar tablero cargado
3. Buscar Llave
4. Buscar Salida
5. Creditos
6. Salir

Introduzca opcion (1-6):

```

*Figura 17: Mostrando el laberinto por pantalla.*

- Luego debemos realizar los pasos 3 para buscar el camino hacia la llave y en seguida colocar 4 para buscar el camino hacia la salida, tener en cuenta que, si uno revisa el archivo creado antes de terminar las dos opciones, solo se verá el camino que solicito buscar, como se aprecia en la Figura 18:

```

1. Cargar Tablero
2. Mostrar tablero cargado
3. Buscar Llave
4. Buscar Salida
5. Creditos
6. Salir

Introduzca opcion (1-6): 3

1. Cargar Tablero
2. Mostrar tablero cargado
3. Buscar Llave
4. Buscar Salida
5. Creditos
6. Salir

Introduzca opcion (1-6): 4

1. Cargar Tablero
2. Mostrar tablero cargado
3. Buscar Llave
4. Buscar Salida
5. Creditos
6. Salir

Introduzca opcion (1-6): █

```

*Figura 18: Buscando camino hacia la Llave y luego hacia la salida.*

- Finalmente, para salir del software se selecciona la opción número 6, como muestra la Figura 19:

```

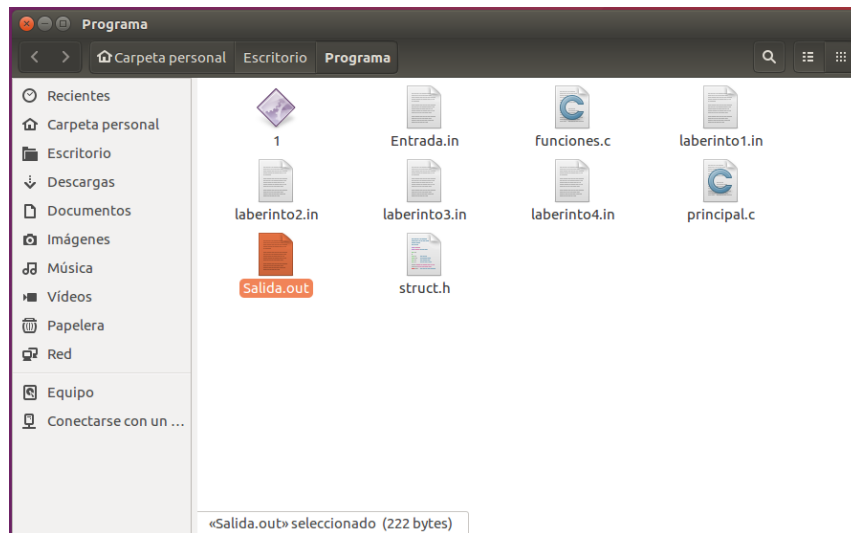
1. Cargar Tablero
2. Mostrar tablero cargado
3. Buscar Llave
4. Buscar Salida
5. Creditos
6. Salir

Introduzca opcion (1-6): 6
huaso@huaso-VirtualBox:~/Escritorio/Programa$ █

```

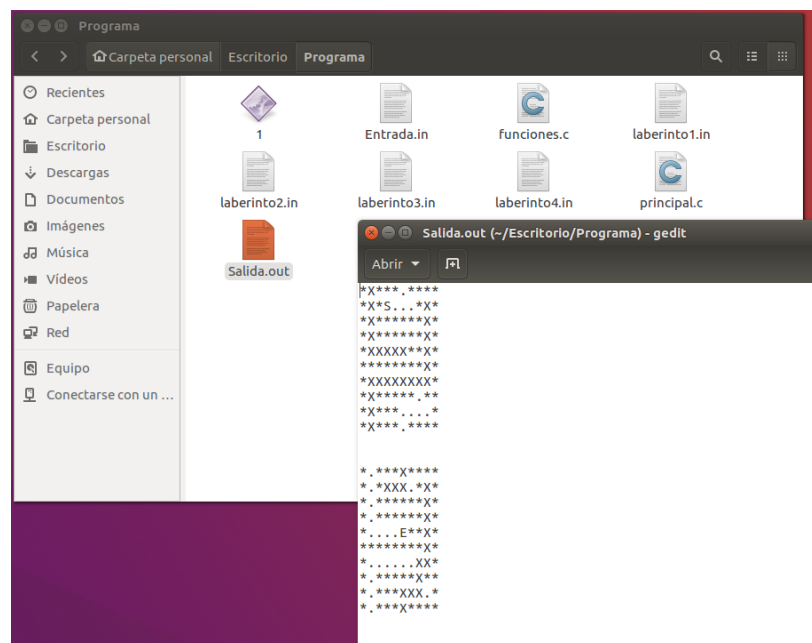
*Figura 19: Cerrando el software.*

- Luego el software se cierra y si nos dirigimos a la carpeta donde tenemos almacenado el programa no aparece el archivo con la salida, como se puede observar en la Figura 20:



*Figura 20: Carpeta con el archivo de “salida.out” del software*

- Luego abrimos el archivo para verificar la solución final del “Laberinto” que entrega el software, como muestra la Figura 21:



*Figura 21: Salida final del software.*



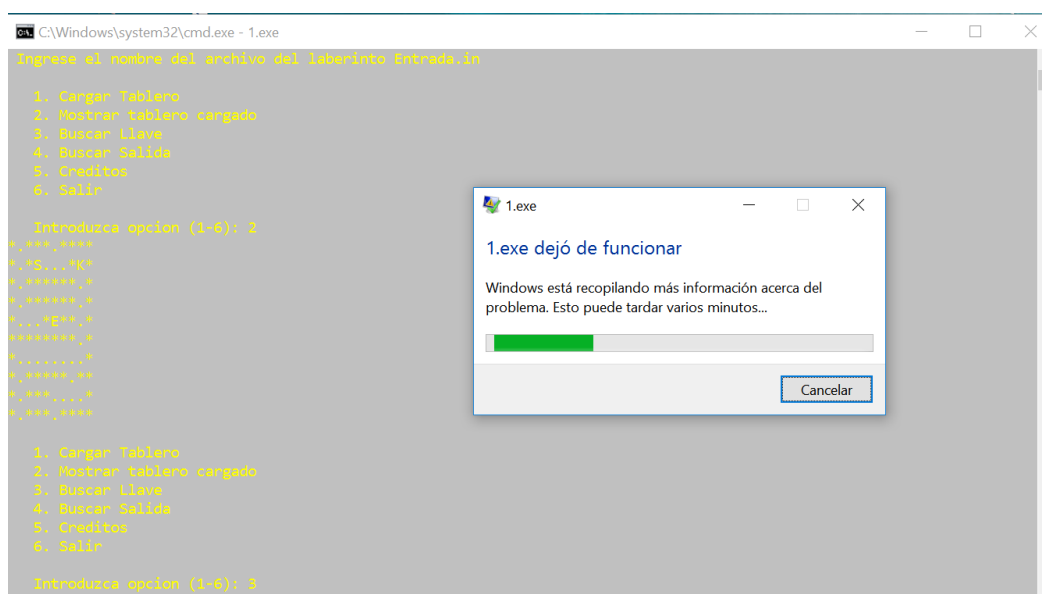
## CAPÍTULO 4. EXITOS Y POSIBLES ERRORES

En ejecución la aplicación paso todas las pruebas que se le realizaron, tener en cuenta que fue sometida alrededor a distintos “Laberintos” teniendo siempre un resultado exitoso.

Algunos de los errores posibles que puede hacer que la aplicación son los siguientes:

- Cuando el “Laberinto” no cuenta con una salida, entrada o llave dentro de su mapa.
- Cuando el “Laberinto” no tiene un camino posible para poder llegar a uno de los objetivos mencionados en el punto anterior.

En la Figura 22, se puede observar como el software deja de funcionar cuando se encuentra con uno de los errores mencionados:



*Figura 22: Posible error dentro de software.*