

# **Estructura de Datos y Análisis de Algoritmos Manual de Usuario Laboratorio 1**

**Ariel Ignacio Tirado Maturana**

Profesor:  
Jacqueline Kohler Casasempere  
Alejandro Cisterna Villalobos

Ayudante:  
Gerardo Zuñiga Leal

Santiago - Chile  
2-2016



## TABLA DE CONTENIDOS

Tabla de Contenidos.....	I
Índice de Figuras .....	I
CAPÍTULO 1. Introducción .....	1
CAPÍTULO 2. Como compilar y ejecutar .....	2
2.1 Linux (Ubuntu).....	2
2.1.1 Compilar.....	2
2.1.2 Ejecutar.....	2
2.2 Windows .....	3
2.2.1 Compilar.....	3
2.2.2 Ejecutar.....	4
2.3 Entradas y salidas del programa .....	5
CAPÍTULO 3. Posibles Errores.....	7

## ÍNDICE DE FIGURAS

<i>Imagen 2-1: Comando para compilar en Linux.</i> .....	2
<i>Imagen 2-2: Comando para ejecutar en Linux</i> .....	2
<i>Imagen 2-3: Menú del programa en Linux.</i> .....	3
<i>Imagen 2-4: Ejecución exitosa del programa en Linux.</i> .....	3
<i>Imagen 2-5: Acceder a la carpeta en Windows.</i> .....	3
<i>Imagen 2-6: Compilar el programa en Windows.</i> .....	4
<i>Imagen 2-7: Ejecutar el programa en Windows.</i> .....	4
<i>Imagen 2-8: Menú del programa en Windows.</i> .....	4
<i>Imagen 2-9: Ejecución exitosa del programa en Windows.</i> .....	5
<i>Imagen 2-10: Sub-imágenes a comparar.</i> .....	5
<i>Imagen 2-11: Imagen principal.</i> .....	5
<i>Imagen 2-12: Archivo generado con los resultados de las comparaciones.</i> .....	6



## **CAPÍTULO 1. INTRODUCCIÓN**

En el presente manual de usuario, se le indicará a la persona que utilice el programa adjunto como ejecutarlo, que cosas debe tener en consideración a la hora de su ejecución y se le indicarán los posibles errores a la hora de ejecutarlo.

El programa solicitado a los alumnos del curso de Análisis de Algoritmos y Estructuras de Datos para su primera entrega de laboratorio consiste en la creación de un algoritmo, escrito en el lenguaje de programación C, el cual consiste en comparar imágenes dispuestas en un formato de pixeles RGB en dos archivos de texto, los cuales funcionarán como entradas para el programa y verificar si existen o no coincidencias entre ellas, generando así un tercer archivo de texto con el resultado de estas comparaciones.

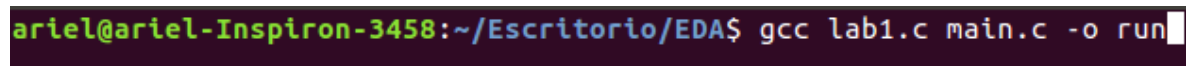
## CAPÍTULO 2. COMO COMPILAR Y EJECUTAR

### 2.1 LINUX (UBUNTU)

#### 2.1.1 Compilar

Para compilar en alguna distribución de Linux, más específicamente Ubuntu, debe acceder a la terminal del Sistema operativo y acceder desde ahí a la carpeta donde están ubicados los archivos *lab1.c*, *main.c* y *headers.h*. Para esto debe ingresar el comando *cd* seguido de la dirección de la carpeta almacenadora (Cuando se inicia la terminal, por defecto se está en la raíz del Sistema Operativo).

Luego de realizar este procedimiento, se debe proceder a compilar los archivos. Para esto debe ingresar el comando indicado en la siguiente imagen:



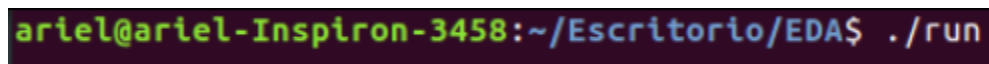
```
ariel@ariel-Inspiron-3458:~/Escritorio/EDA$ gcc lab1.c main.c -o run
```

*Imagen 2-1: Comando para compilar en Linux.*

Al presionar el botón *Enter* se compilará el programa y se generará un archivo ejecutable llamado *run*.

#### 2.1.2 Ejecutar

Luego de haber compilado el programa, se debe proceder a ejecutarlo. Para esto se debe escribir en la línea de comandos el nombre que se le dio al ejecutable a la hora de compilar, como se muestra en la imagen:



```
ariel@ariel-Inspiron-3458:~/Escritorio/EDA$ ./run
```

*Imagen 2-2: Comando para ejecutar en Linux*

Luego de presionar el botón *Enter*, se abrirá el menú del programa:

```
artel@artel-Inspiron-3458:~/Escritorio/EDAS ./run
||-----||
|| Recordatorio: Cada vez que ingrese el nombre de un archivo, procure ingresar tambien su extension, por ejemplo, nombre_archivo.txt ||
||-----||
|| Ingrese el nombre del archivo que contiene la imagen principal: ||
```

*Imagen 2-3: Menú del programa en Linux.*

En este menú se le solicitará al usuario que ingrese los nombres de los archivos de entrada, incluyendo la extensión del archivo. Luego de escribir ambos nombres, el programa ejecutará el algoritmo, entregando la solución en el tercer archivo, llamado *resultado.txt*, el cual será almacenado en la carpeta en la cual se encuentra el código fuente. La siguiente imagen ilustra lo mencionado anteriormente:

```
||-----||
|| Recordatorio: Cada vez que ingrese el nombre de un archivo, procure ingresar tambien su extension, por ejemplo, nombre_archivo.txt ||
||-----||
|| Ingrese el nombre del archivo que contiene la imagen principal: imagenPrincipal.txt ||
|| Ingrese el nombre del archivo que contiene las imagenes a comparar: ImagesBuscar.txt ||
||-----||
El algoritmo demora un total de 0.000000 segundos en ejecutarse.
```

*Imagen 2-4: Ejecución exitosa del programa en Linux.*

## 2.2 WINDOWS

### 2.2.1 Compilar

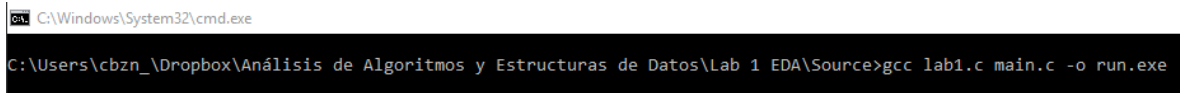
Para compilar en Windows, debe acceder al Símbolo del Sistema, ingresando la línea *cmd* en el buscador del Sistema Operativo. Luego de haber ingresado al Símbolo del Sistema, debe buscar la carpeta donde están ubicados los archivos *main.c*, *lab1.c* y *headers.h*, esto se logra usando el comando *cd* seguido de la dirección de la carpeta almacenadora.

```
C:\> Símbolo del sistema
Microsoft Windows [Versión 10.0.10586]
(c) 2015 Microsoft Corporation. Todos los derechos reservados.

C:\Users\cbzn_>cd "C:\Users\cbzn_\Dropbox\Análisis de Algoritmos y Estructuras de Datos\Lab 1 EDA"
C:\Users\cbzn_\Dropbox\Análisis de Algoritmos y Estructuras de Datos\Lab 1 EDA>
```

*Imagen 2-5: Acceder a la carpeta en Windows.*

Una vez se encuentre en la carpeta, se debe compilar el archivo usando el comando indicado en la siguiente imagen:



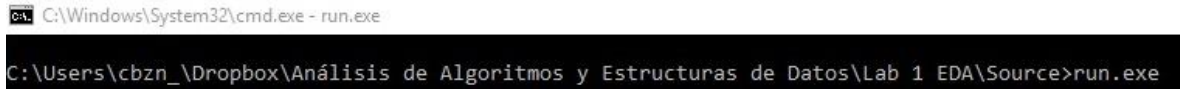
```
C:\Windows\System32\cmd.exe
C:\Users\cbzn_\Dropbox\Análisis de Algoritmos y Estructuras de Datos\Lab 1 EDA\Source>gcc lab1.c main.c -o run.exe
```

*Imagen 2-6: Compilar el programa en Windows.*

Al presionar el botón *Enter* los archivos se compilarán y se creará un archivo ejecutable llamado *run.exe*.

### 2.2.2 Ejecutar

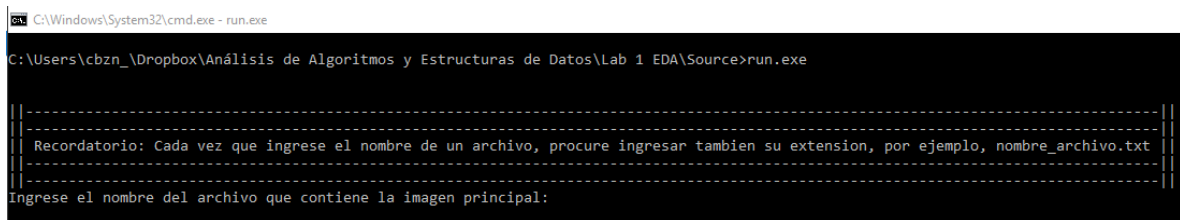
Luego de haber compilado los archivos, el programa estará listo para ser ejecutado. Para realiza este paso, el usuario debe ejecutar el archivo ejecutable creado anteriormente, como se muestra en la siguiente imagen:



```
C:\Windows\System32\cmd.exe - run.exe
C:\Users\cbzn_\Dropbox\Análisis de Algoritmos y Estructuras de Datos\Lab 1 EDA\Source>run.exe
```

*Imagen 2-7: Ejecutar el programa en Windows.*

Luego de presionar *Enter* se ejecutará el programa, mostrando el menú al usuario:



```
C:\Windows\System32\cmd.exe - run.exe
C:\Users\cbzn_\Dropbox\Análisis de Algoritmos y Estructuras de Datos\Lab 1 EDA\Source>run.exe

||-----||
|| Recordatorio: Cada vez que ingrese el nombre de un archivo, procure ingresar tambien su extension, por ejemplo, nombre_archivo.txt ||
||-----||
|| Ingrese el nombre del archivo que contiene la imagen principal: ||
||-----||
```

*Imagen 2-8: Menú del programa en Windows.*

En este paso, el usuario deberá ingresar los nombres de los archivos de entrada, incluyendo la extensión de estos. Luego de haber ingresado los nombres, el algoritmo se ejecutará y creará un archivo de texto llamado *resultado.txt* en la carpeta donde se encuentra el código



fuente, donde se indicarán las imágenes que fueron encontradas y las que no lo fueron. La siguiente imagen muestra la ejecución del presente programa:

```

C:\Windows\System32\cmd.exe
C:\Users\cbzn_\Dropbox\Análisis de Algoritmos y Estructuras de Datos\Lab 1 EDA\Source>run.exe

-----
Recordatorio: Cada vez que ingrese el nombre de un archivo, procure ingresar tambien su extension, por ejemplo, nombre_archivo.txt
-----
Ingrese el nombre del archivo que contiene la imagen principal: imagenPrincipal.txt
Ingrese el nombre del archivo que contiene las imagenes a comparar: imagesBuscar.txt
El algoritmo demoro un total de 0.000000 segundos en ejecutarse.

```

*Imagen 2-9: Ejecución exitosa del programa en Windows.*

## 2.3 ENTRADAS Y SALIDAS DEL PROGRAMA

Se consideran los siguientes archivos de texto como ejemplo para mostrar los resultados del programa.

```

imagesBuscar.txt
3 3
255,255,255 255,255,255 255,255,255
255,255,255 0,0,0 255,255,255
255,255,255 255,255,255 255,255,255
-----
3 3
255,255,255 0,0,0 255,255,255
0,0,0 255,255,255 0,0,0
255,255,255 0,0,0 255,255,255
-----
2 2
255,255,255 0,0,0
255,255,255 255,255,255

```

*Imagen 2-10: Sub-imágenes a comparar.*

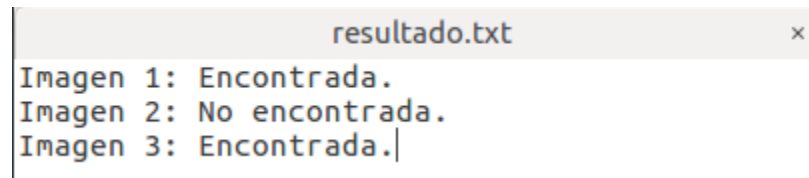
```

imagenPrincipal.txt
5 5
0,0,0 0,0,0 0,0,0 0,0,0 0,0,0
0,0,0 0,0,0 0,0,0 0,0,0 0,0,0
0,0,0 255,255,255 255,255,255 255,255,255 0,0,0
0,0,0 255,255,255 0,0,0 255,255,255 0,0,0
0,0,0 255,255,255 255,255,255 255,255,255 0,0,0

```

*Imagen 2-11: Imagen principal.*

Donde se generará la siguiente salida:



```
resultado.txt
Imagen 1: Encontrada.
Imagen 2: No encontrada.
Imagen 3: Encontrada.
```

*Imagen 2-12: Archivo generado con los resultados de las comparaciones.*

## **CAPÍTULO 3. POSIBLES ERRORES**

A continuación se indicará al usuario un listado de posibles errores que pueden ocurrir con el uso del programa:

- Los archivos de entrada deben poseer un formato definido. En caso de ingresar archivos con un formato erróneo, se pueden originar problemas a la hora de leer los datos, generando pérdidas de la memoria del computador y en muchos casos, el apagado forzado del ordenador.
- Al ingresar los nombres de los archivos de entrada, debe procurar escribirlos cuidadosamente y no intercambiar los nombres entre el archivo que contiene la imagen principal y las sub-imágenes, ya que los formatos de lectura de ambos poseen diferencias.