



**UNIVERSIDAD DE SANTIAGO DE CHILE  
FACULTAD DE INGENIERÍA  
DEPARTAMENTO DE INGENIERÍA INFORMÁTICA**

## **Estructura de Datos y Análisis de Algoritmos**

### **Manual de Usuario**

Alumno: Israel Arias Panez.

Profesor: Mario Inostroza.  
Ayudante: Esteban Silva.

Santiago - Chile  
2-2020

## MANUAL DE USUARIO

### 3.1 INTRODUCCIÓN

El programa que se encuentra a su disposición le permite generar todas las posibles tripulaciones que pueden abordar la nave que busca emigrar del planeta Tierra por motivo de la expansión del virus Covid-19 de acuerdo con la información de los tripulantes que ingrese al programa, incluso se le mostrará ordenadamente la mejor tripulación de acuerdo con su calificación hasta la peor tripulación. En este manual de usuario se le indicará las instrucciones de uso del programa, entradas, condiciones para que el programa funcione, entre otros.

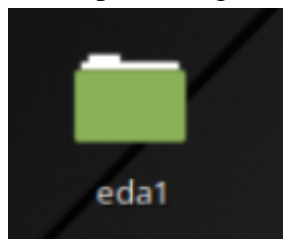
El programa se encuentra escrito en el lenguaje C, cuyo código se encuentra a su disposición.

### 3.2 CÓMO COMPILAR Y EJECUTAR

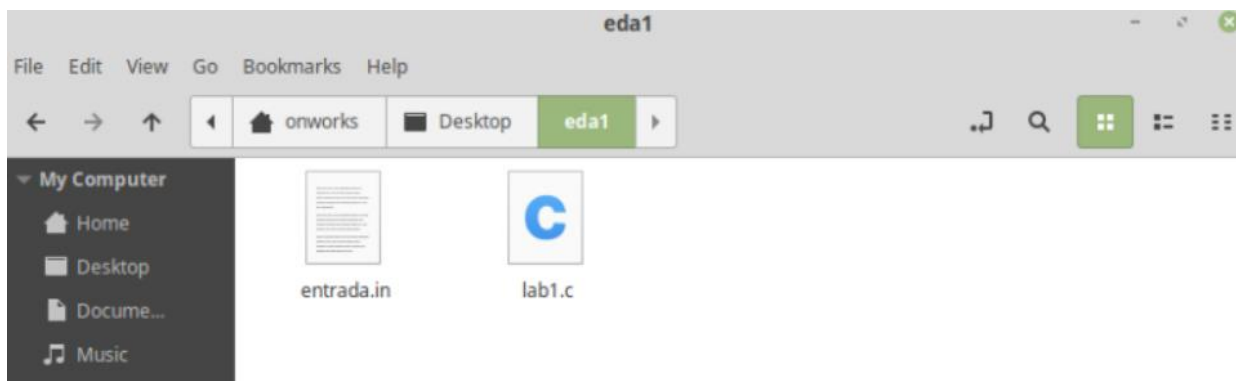
En esta sección se detallará, paso a paso como compilar y ejecutar el programa en dos sistemas operativos distintos, Linux y Windows.

#### 3.2.1 Compilación en Linux

En primer lugar, se debe verificar que se tengan todos los archivos necesarios en una carpeta



Se tomará como ejemplo en este caso la carpeta eda1, la carpeta puede tener el nombre que guste.

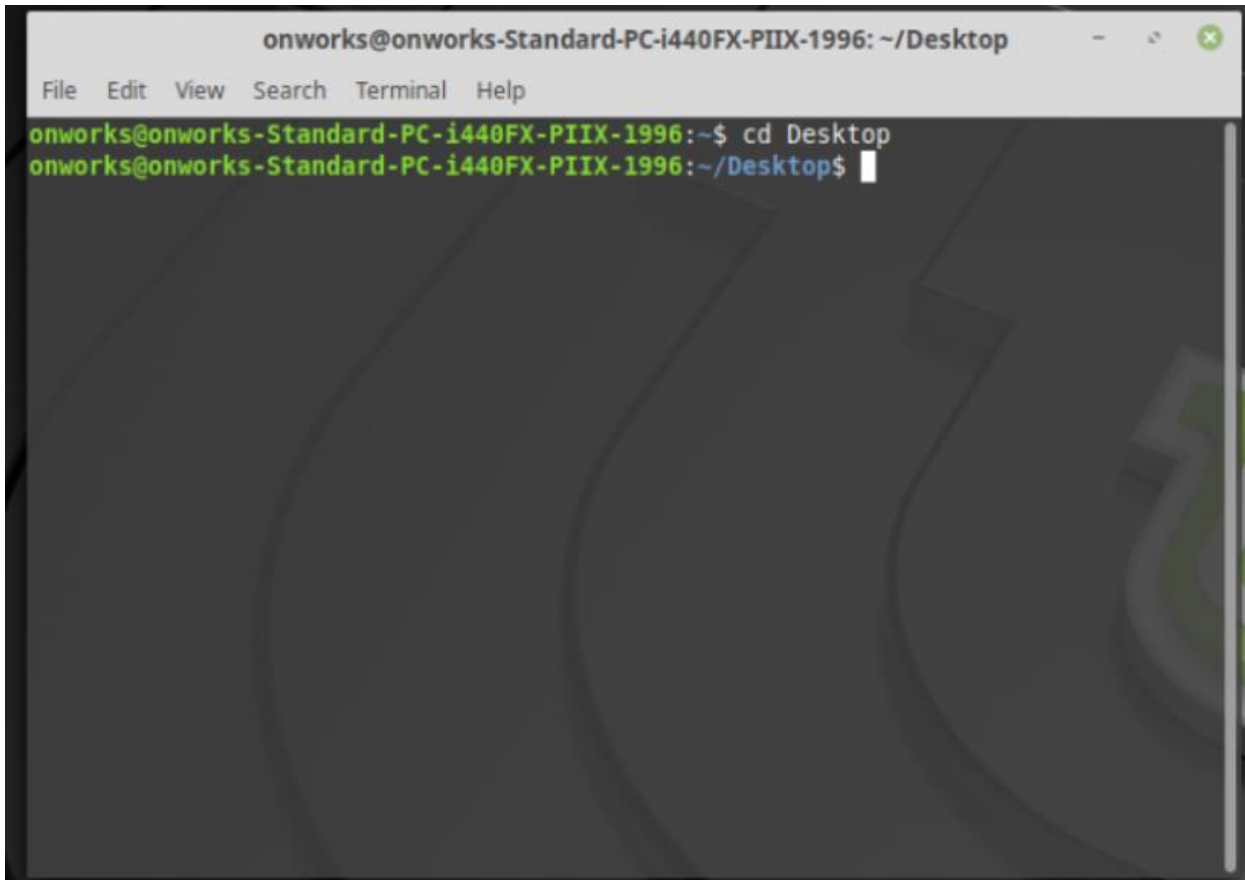


Se debe verificar que se tenga el programa lab1.c y la entrada.in necesaria para la ejecución del programa.

A continuación, se debe abrir la terminal, para esto basta con hacer click en la terminal abajo en la barra de herramientas.



Se debería abrir la terminal, en la cual necesitamos ir al directorio o carpeta en el cual se tiene el programa, en este caso se encuentra en `eda1`, entonces se hace uso del comando `cd Desktop`, para ir a la carpeta `desktop`

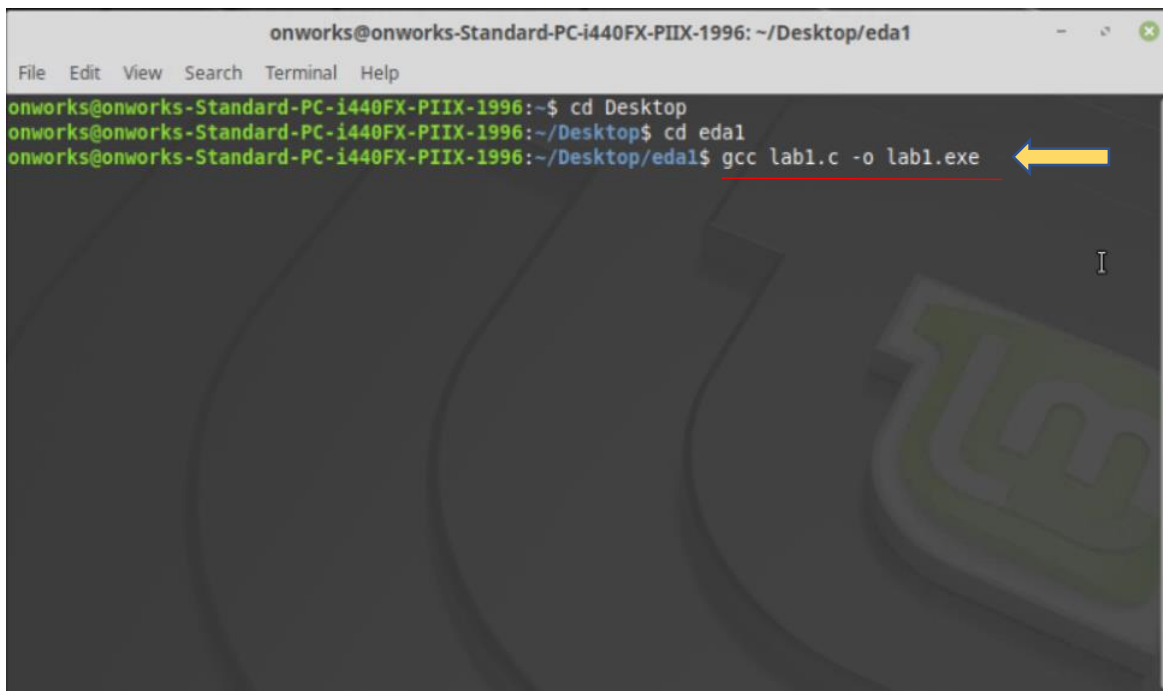


Luego haciendo uso del mismo comando se accede a la carpeta eda1.

A terminal window titled 'onworks@onworks-Standard-PC-i440FX-PIIX-1996: ~/Desktop/eda1'. The window has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The terminal shows three lines of commands: 'onworks@onworks-Standard-PC-i440FX-PIIX-1996:~\$ cd Desktop', 'onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop\$ cd eda1', and 'onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/eda1\$' with a cursor at the end.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996: ~/$ cd Desktop
onworks@onworks-Standard-PC-i440FX-PIIX-1996: ~/Desktop$ cd eda1
onworks@onworks-Standard-PC-i440FX-PIIX-1996: ~/Desktop/eda1$
```

Luego para compilar el programa se usa el siguiente comando

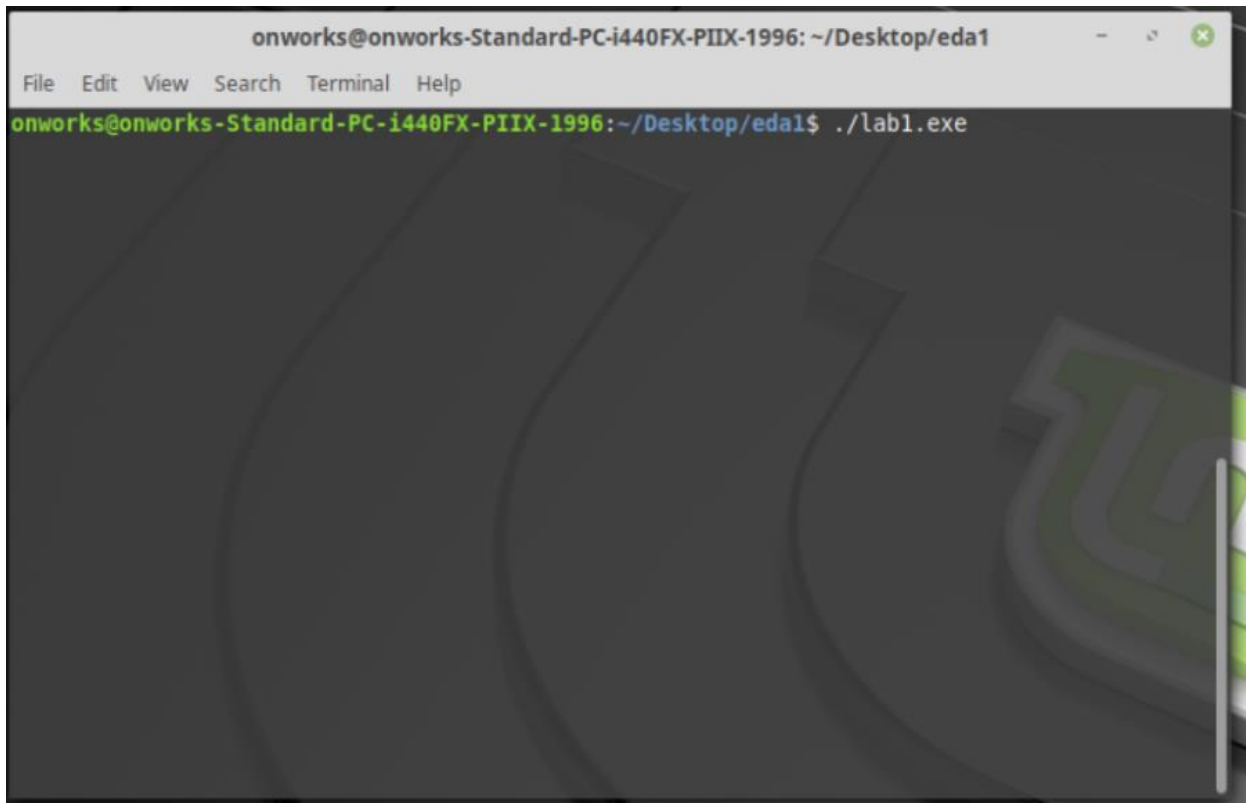
A terminal window titled 'onworks@onworks-Standard-PC-i440FX-PIIX-1996: ~/Desktop/eda1'. The window has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The terminal shows three lines of commands: 'onworks@onworks-Standard-PC-i440FX-PIIX-1996:~\$ cd Desktop', 'onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop\$ cd eda1', and 'onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/eda1\$ gcc lab1.c -o lab1.exe'. A yellow arrow points to the command 'gcc lab1.c -o lab1.exe'.

```
onworks@onworks-Standard-PC-i440FX-PIIX-1996:~$ cd Desktop
onworks@onworks-Standard-PC-i440FX-PIIX-1996: ~/Desktop$ cd eda1
onworks@onworks-Standard-PC-i440FX-PIIX-1996: ~/Desktop/eda1$ gcc lab1.c -o lab1.exe
```

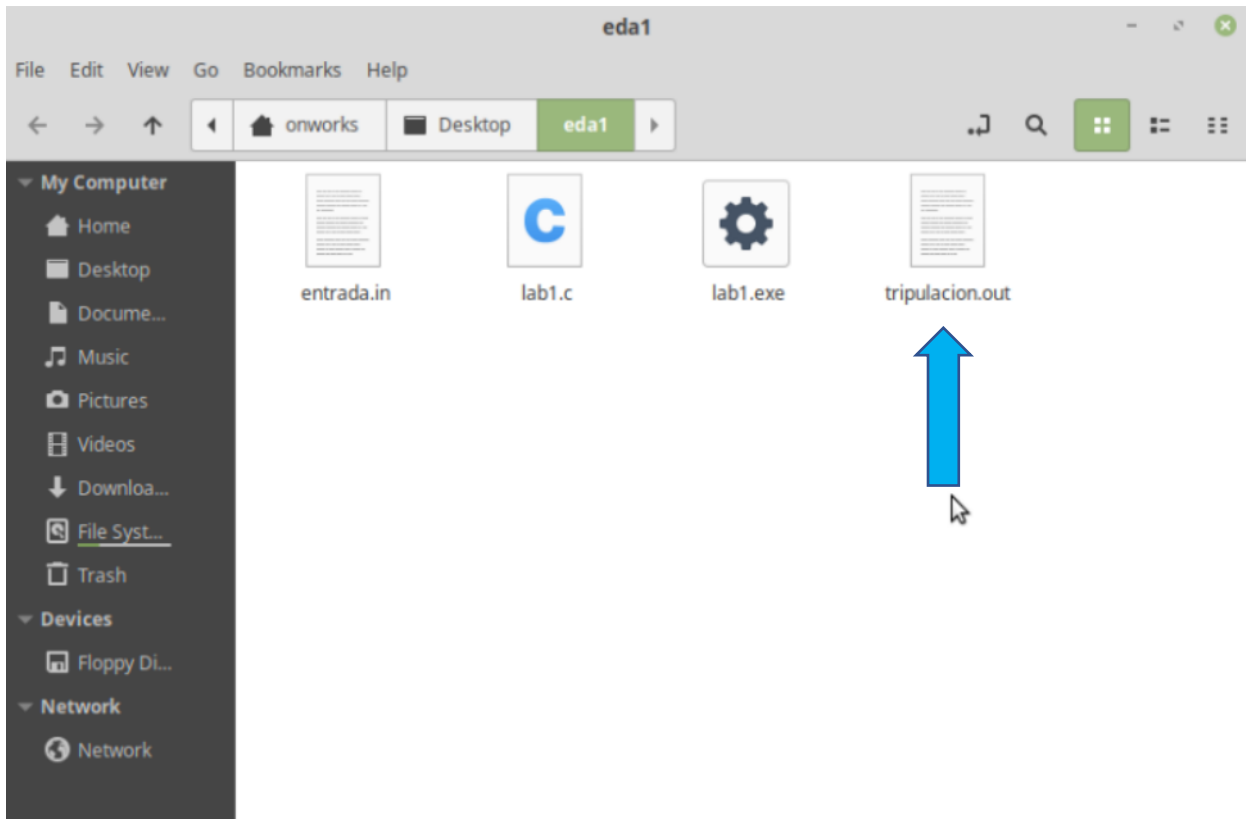
El comando es `gcc lab1.c -o lab1.exe`

// `lab1.c` es el nombre del programa y `lab1.exe` es el nombre que tendrá el ejecutable del programa, no es necesario que el ejecutable se llame `lab1.exe`, puede ser cualquier nombre a elección, sin embargo, si es necesario ingresar `lab1.c` ya que ese es el nombre del programa en su archivo.c

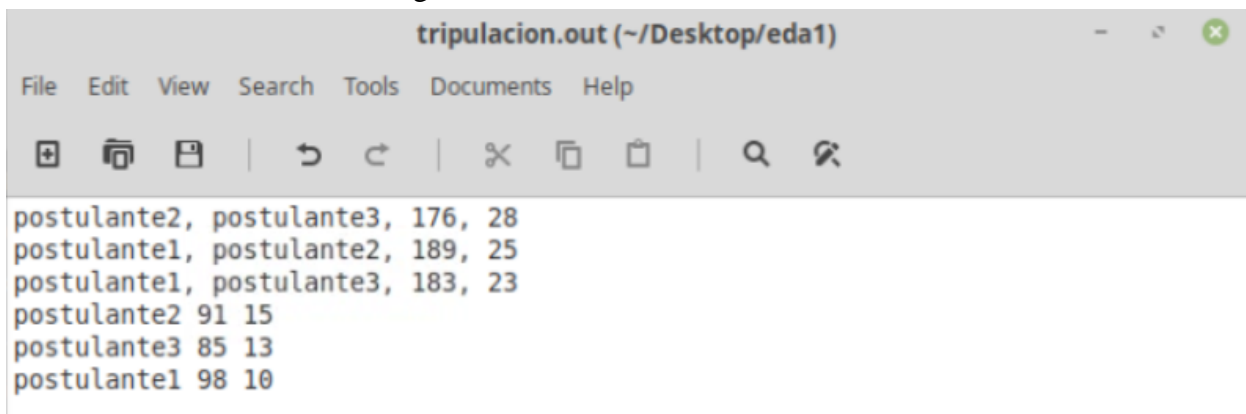
Ahora solo queda ejecutar el archivo `.exe` creado en la compilación anterior, para ello se usa el comando `./lab1.exe` es importante destacar que es necesario el punto y el slash antes de nombrar al ejecutable, si no, no funcionara.

A screenshot of a terminal window titled "onworks@onworks-Standard-PC-i440FX-PIIX-1996: ~/Desktop/eda1". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal text shows the prompt "onworks@onworks-Standard-PC-i440FX-PIIX-1996:~/Desktop/eda1\$" followed by the command ". /lab1.exe" entered on the next line. The background of the terminal is dark with a faint, stylized logo on the right side.

Ahora el programa se debería haber ejecutado sin problemas, entonces se debe volver a abrir la carpeta eda1, debiese verse así:



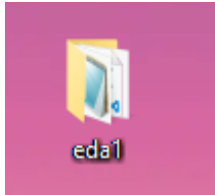
Y al abrir el archivo tripulacion.out generado por el programa, el cual se encuentra señalado con una flecha celeste en la imagen anterior:



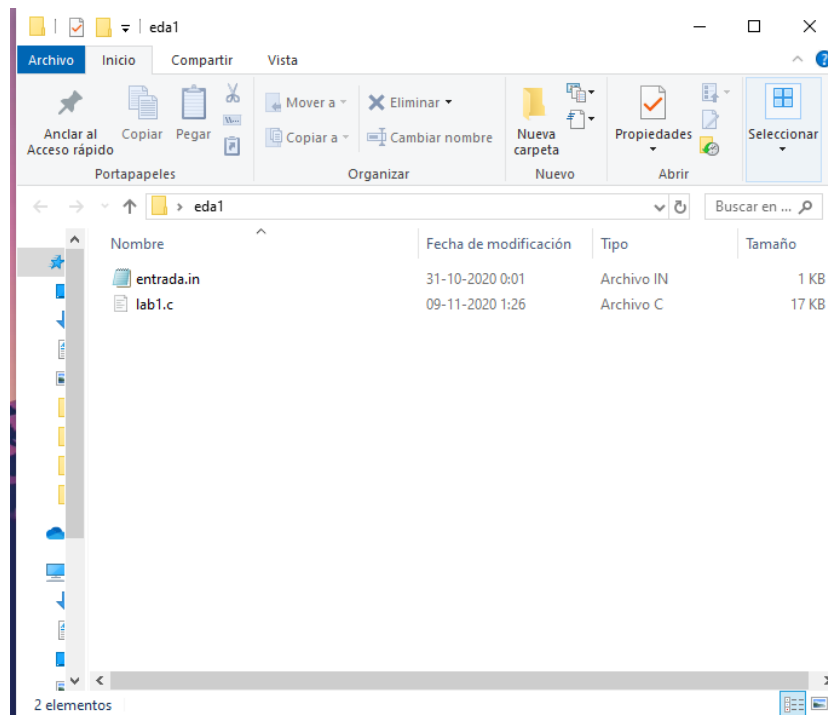
Debiese mostrarse lo que se ve en la imagen, el resultado de la ejecución del programa.

### 3.2.2 Compilación en Windows

En primer lugar, se debe verificar que se tengan todos los archivos necesarios en una carpeta

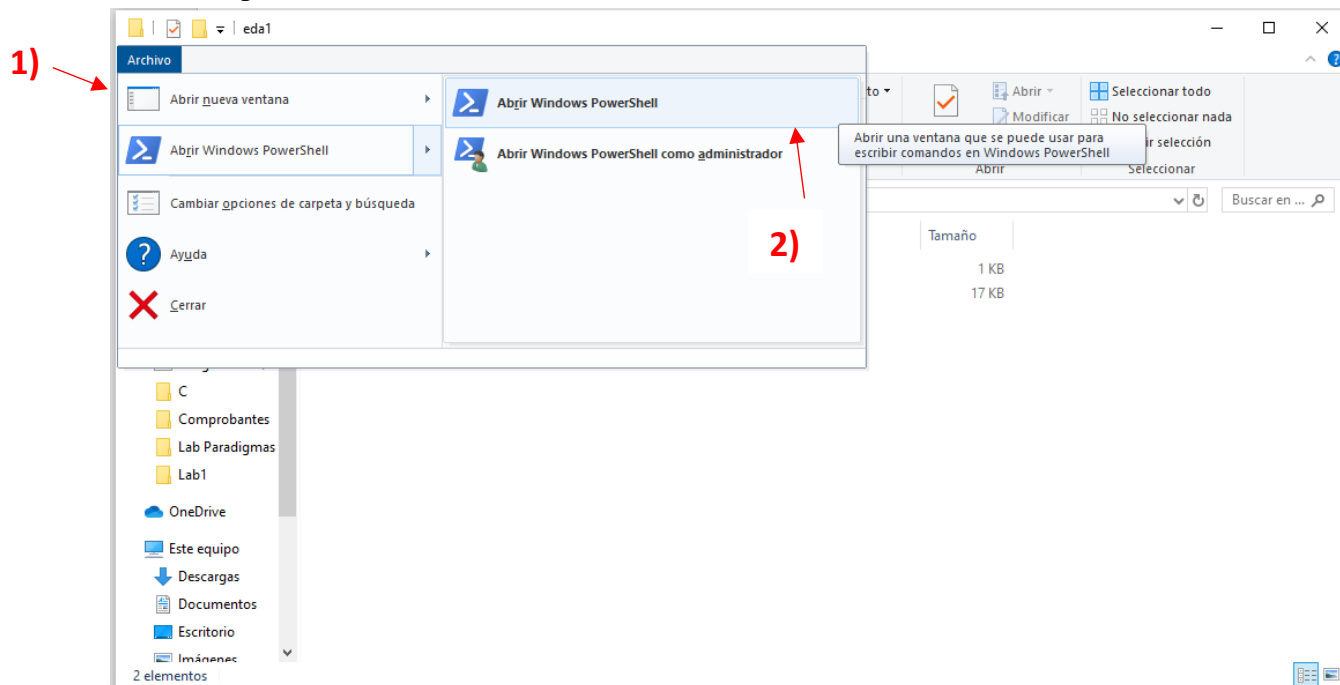


Se tomará como ejemplo en este caso la carpeta eda1, la carpeta puede tener el nombre que guste.

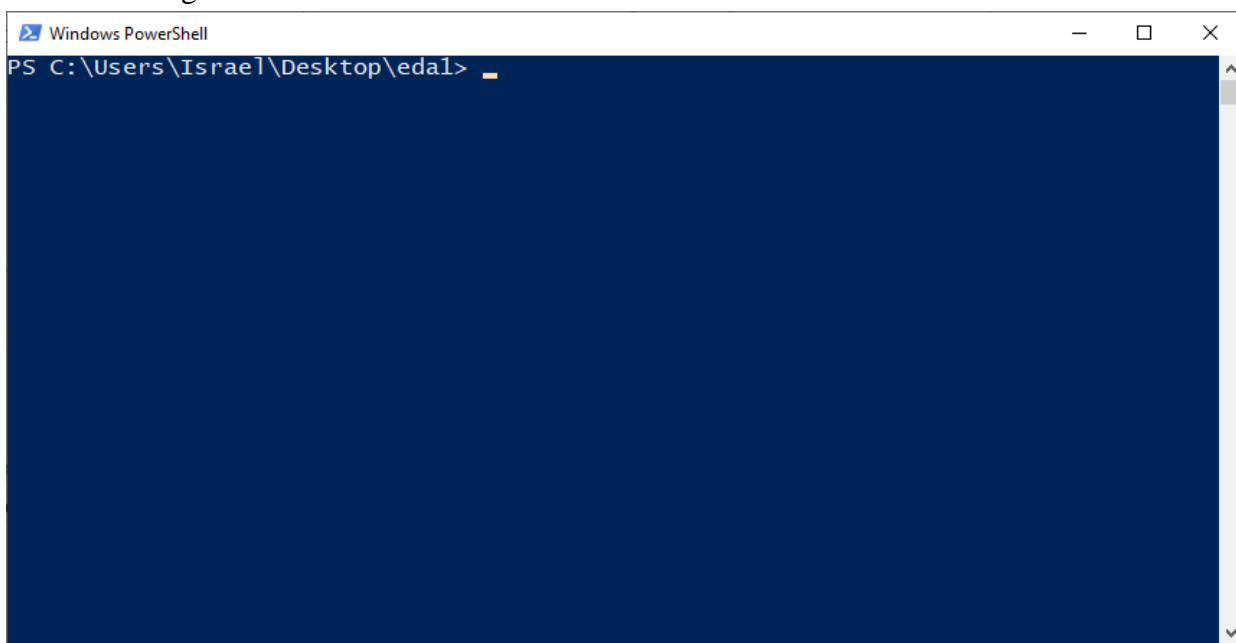


Se debe verificar que se tenga el programa lab1.c y la entrada.in necesaria para la ejecución del programa.

A continuación, se debe presionar en archivo en la esquina superior izquierda, luego abrir Windows PowerShell.

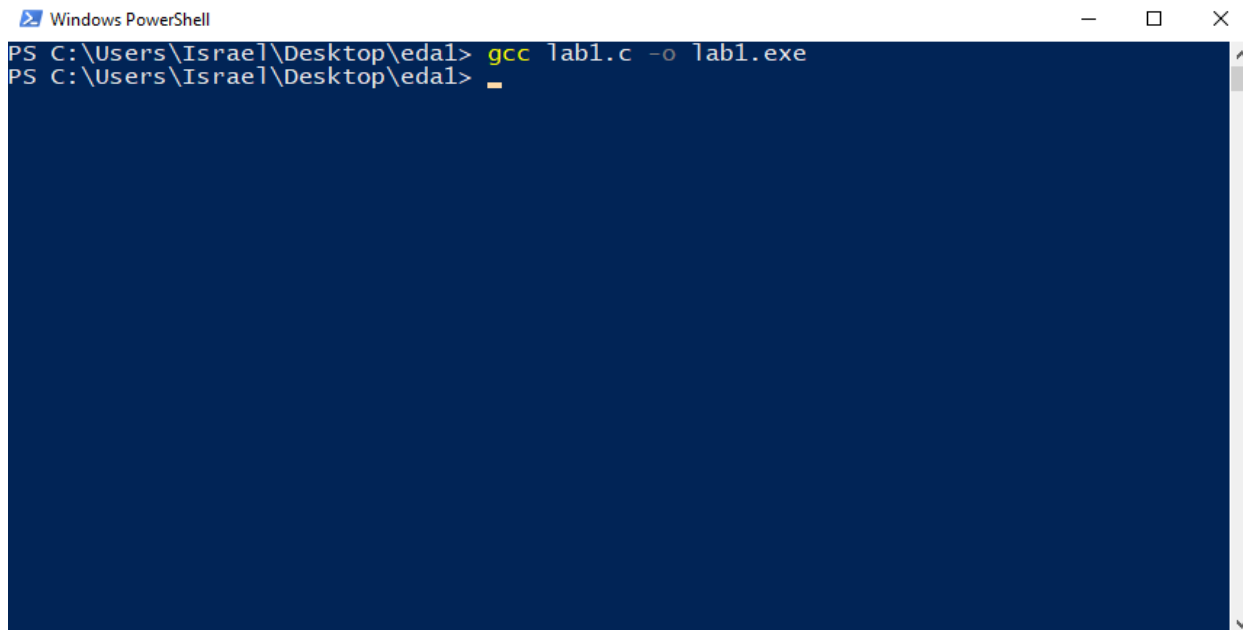


Se abrirá la siguiente ventana





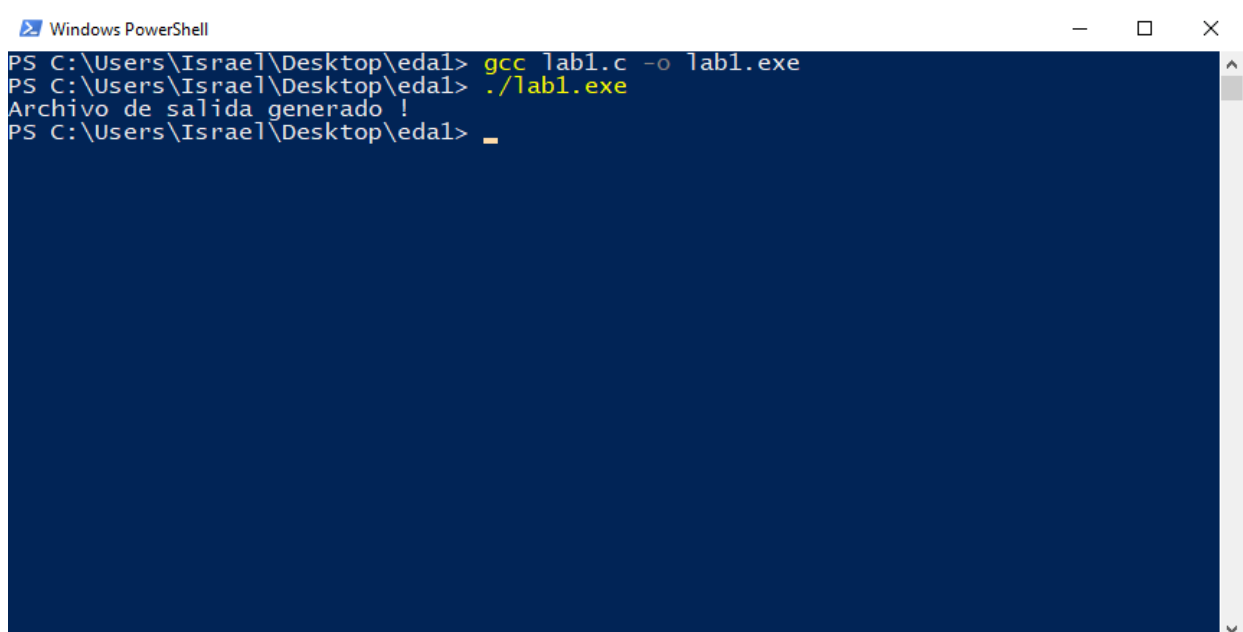
Ahora para compilar el programa se debe usar el comando `gcc lab1.c -o lab1.exe`



```
Windows PowerShell
PS C:\Users\Israel\Desktop\edal> gcc lab1.c -o lab1.exe
PS C:\Users\Israel\Desktop\edal> █
```

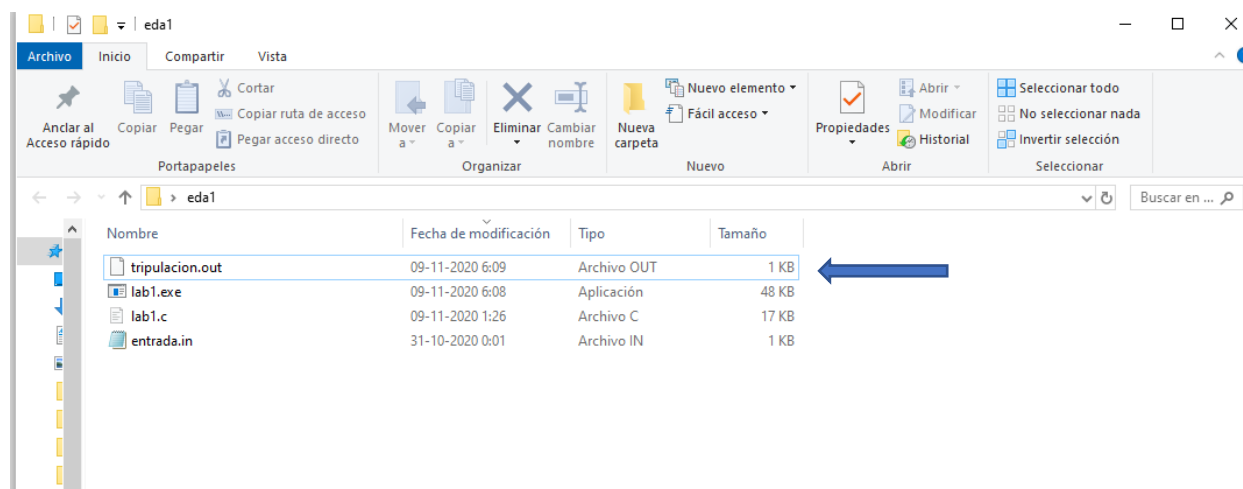
// `lab1.c` es el nombre del programa y `lab1.exe` es el nombre que tendrá el ejecutable del programa, no es necesario que el ejecutable se llame `lab1.exe`, puede ser cualquier nombre a elección, sin embargo, si es necesario ingresar `lab1.c` ya que ese es el nombre del programa en su archivo.c

Ahora solo queda ejecutar el archivo .exe creado en la compilación anterior, para ello se usa el comando `./lab1.exe` es importante destacar que es necesario el punto y el slash antes de nombrar al ejecutable, si no, no funcionara.

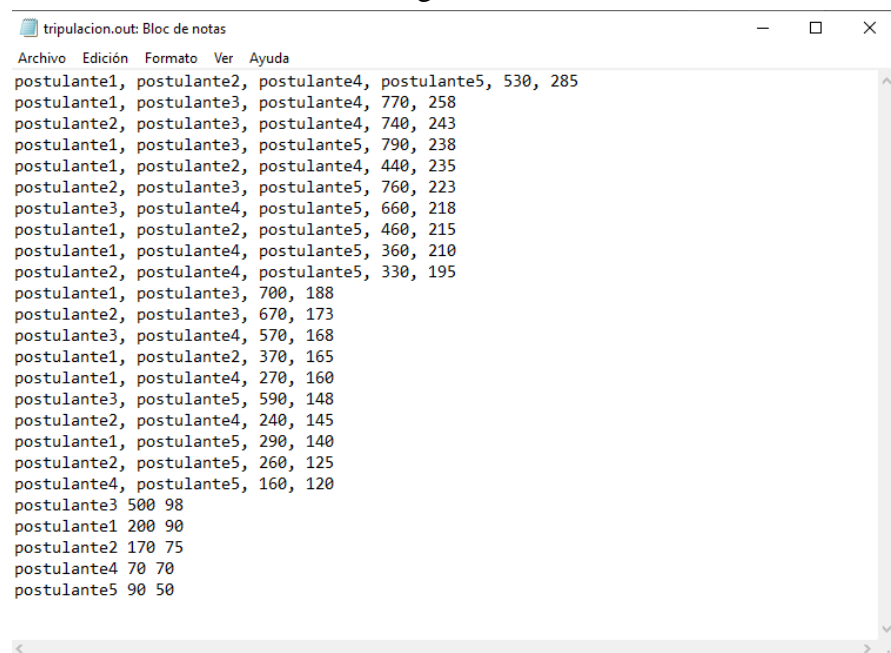


```
Windows PowerShell
PS C:\Users\Israel\Desktop\edal> gcc lab1.c -o lab1.exe
PS C:\Users\Israel\Desktop\edal> ./lab1.exe
Archivo de salida generado !
PS C:\Users\Israel\Desktop\edal> █
```

Ahora el programa se debería haber ejecutado sin problemas, entonces se debe volver a abrir la carpeta eda1, debiese verse así:



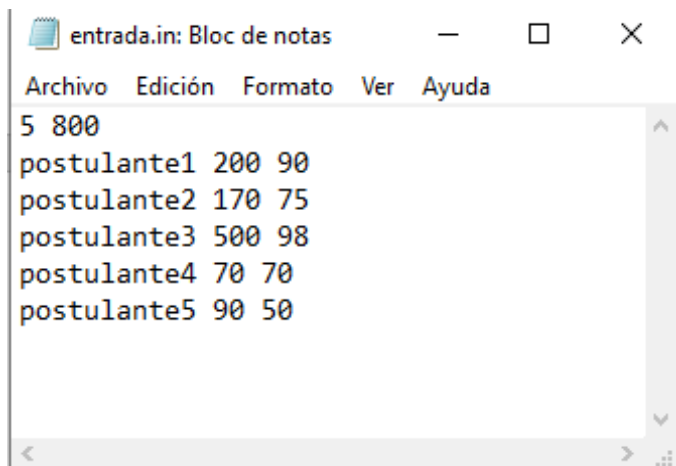
Y al abrir el archivo tripulacion.out generado por el programa, el cual se encuentra señalado con una flecha celeste en la imagen anterior:



Debiese mostrarse lo que se ve en la imagen, el resultado de la ejecución del programa.

### 3.3 FUNCIONALIDADES DEL PROGRAMA

Como se mencionó en la introducción, este programa puede generar cualquier combinación de tripulación que abordara la nave, pero para ello necesita una entrada, la cual debe llamarse “entrada.in”



```
entrada.in: Bloc de notas
Archivo  Edición  Formato  Ver  Ayuda
5 800
postulante1 200 90
postulante2 170 75
postulante3 500 98
postulante4 70 70
postulante5 90 50
```

“entrada.in” tiene las siguientes restricciones: el primer valor corresponde a la cantidad de postulantes que buscan abordar la nave, en el ejemplo se ven 5, a continuación de un espacio se debe indicar el peso máximo que soporta la nave, en el ejemplo se ve el numero 800, luego de un salto de línea se escribe el nombre del postulante, luego de un espacio el peso del postulante y luego de otro espacio la clasificación del postulante obtenido en el examen.

Usted puede generar cualquier tipo de combinación que desee, solo debe editar o crear un nuevo archivo, pero siempre debe llamarse “entrada.in” y cumplir con los espacios, saltos de línea y que los primeros dos números siempre sean el numero de postulantes y el peso máximo de la nave.

En caso de no cumplir el formato es probable que el programa no logre ejecutarse adecuadamente o no se ejecute directamente. La salida siempre se llamará “tripulacion.out” y se encontrara en el mismo directorio en el cual se ejecutó el programa.

### **3.4 POSIBLES ERRORES**

Un posible error del programa es el de que el nombre del archivo no sea “entrada.in”, en el caso que eso suceda, el mismo programa debería indicarle que verificara el nombre del archivo, entonces para solucionarlo, usted debe renombrar el archivo de entrada como “entrada.in”.

Otro error común puede ser el que se haya intentado ejecutar el programa dándole como parámetro cero tripulantes, también en esta instancia el programa debería indicar que se intentó hacer una combinación sin tripulantes, lo cual no es posible, para arreglar este problema, basta con verificar y editar el primer valor de “entrada.in” y cambiarlo de cero, así el programa leerá que no son cero postulantes los que intentan abordar la nave.

¿El programa deja de ejecutarse repentinamente y no genera el archivo de salida?

Eso se debe a un mal formato el archivo “entrada.in”, para solucionarlo debe respetarse el formato de escritura del archivo “entrada.in”, el cual se encuentra detallado en el punto 3.3.

### **Limitaciones**

La principal limitación que tiene el programa es que tiene una memoria fija de 100 caracteres para leer el nombre de los tripulantes, en caso de que el nombre del tripulante tuviera más de 100 caracteres, el programa no podría ejecutarse ya que no sería capaz de almacenar el nombre de ese tripulante, esta limitación se da en ambos Sistemas operativos, Windows y Linux.