

# Cuenta Palabras



Parisi Bruno – Suarez Ezequiel

Dado un directorio que contiene una cantidad arbitraria de archivos con extensión .txt el programa procesara cada uno de los archivos y contara la aparición de cada una de las palabras dando como resultado dos archivos adicionales .out denominados cadauno.out y total.out, donde cadauno.out almacenara los respectivos archivos .txt y la cantidad de apariciones de las palabras en cada archivo y totales.out almacenará la apariciones total de cada palabra.

Universidad Nacional del Sur  
Organización de computadoras  
1º cuatrimestre

02/06/2023



## Instructivo

Este instructivo asume que el lector tiene conocimientos previos sobre programación, asimismo se contempla la posibilidad que no hayan ejecutado programas mediante la consola.

### ➤ Como acceder al programa

- 1) Descomprimir el archivo comprimido .rar o .zip.



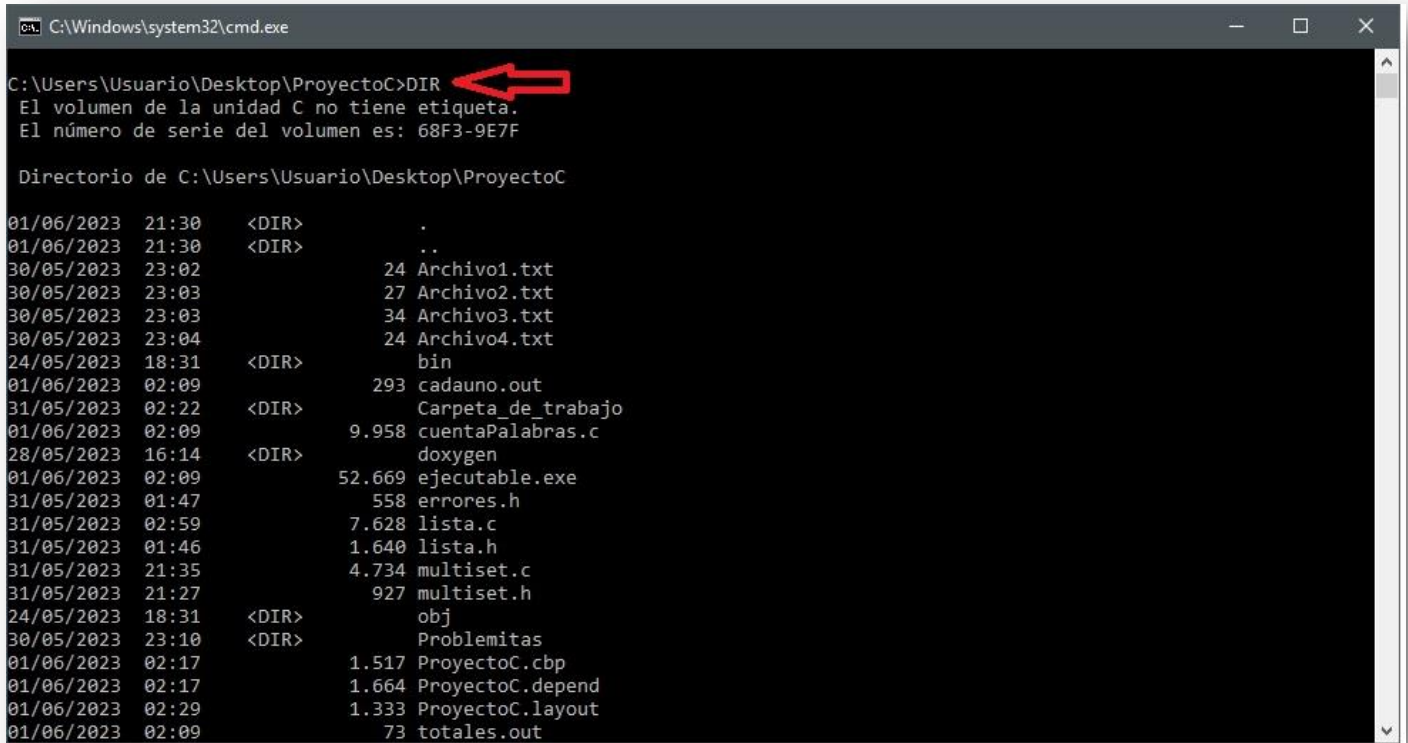
- 2) Acceder a la carpeta descompromida en este caso denominada proyectoC y dirigirse al apartado de “Archivo” y luego “Abrir símbolo de sistema”.



(Agregar acceso mediante comandos cd.. desde cualquier punto del sistema)

3) Una vez abierta la consola nos aparecerá la ruta del directorio en el cual no encontramos.

- Con el comando DIR (provisto por windows) nos listara todo el contenido de la carpeta “proyectoC”.



```
C:\Windows\system32\cmd.exe
C:\Users\Usuario\Desktop\ProyectoC>DIR
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 68F3-9E7F

Directorio de C:\Users\Usuario\Desktop\ProyectoC

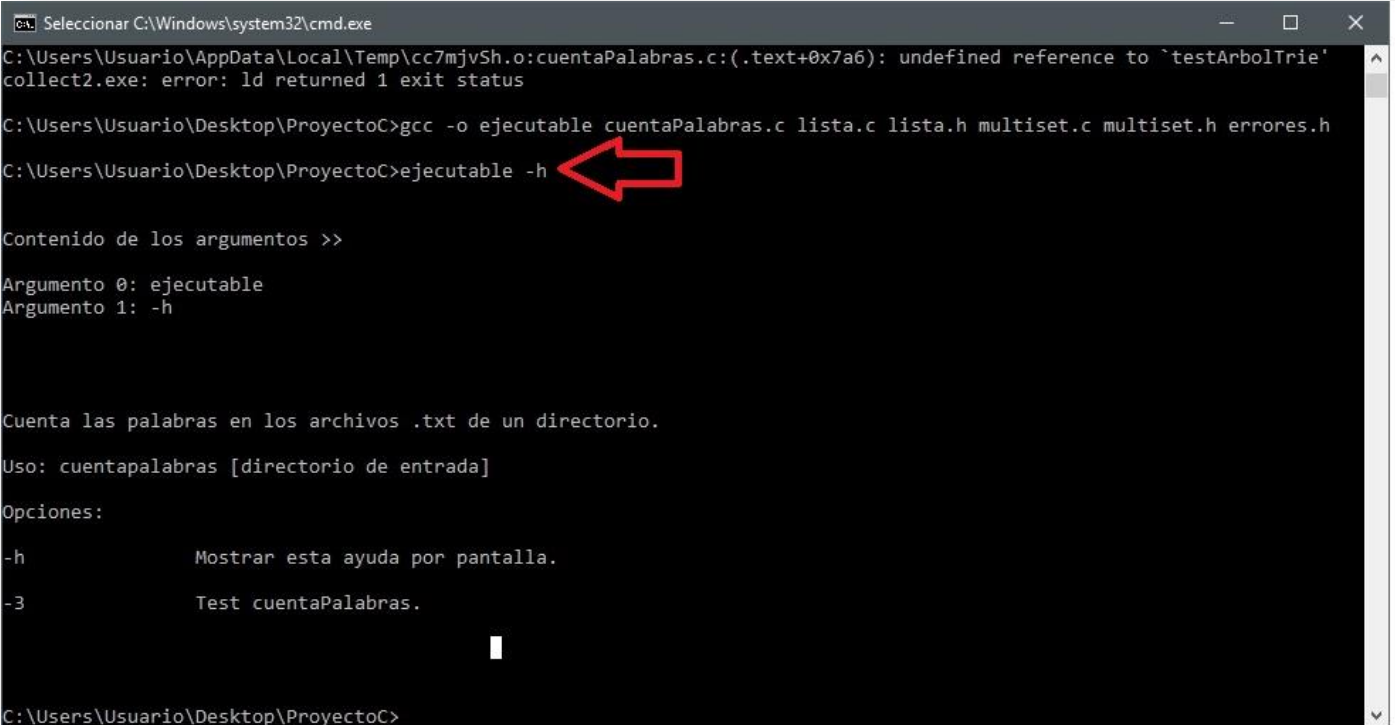
01/06/2023  21:30  <DIR>          .
01/06/2023  21:30  <DIR>          ..
30/05/2023  23:02              24 Archivo1.txt
30/05/2023  23:03              27 Archivo2.txt
30/05/2023  23:03              34 Archivo3.txt
30/05/2023  23:04              24 Archivo4.txt
24/05/2023  18:31  <DIR>          bin
01/06/2023  02:09          293 cadauno.out
31/05/2023  02:22  <DIR>          Carpeta_de_trabajo
01/06/2023  02:09          9.958 cuentaPalabras.c
28/05/2023  16:14  <DIR>          doxygen
01/06/2023  02:09      52.669 ejecutable.exe
31/05/2023  01:47          558 errores.h
31/05/2023  02:59          7.628 lista.c
31/05/2023  01:46          1.640 lista.h
31/05/2023  21:35          4.734 multiset.c
31/05/2023  21:27          927 multiset.h
24/05/2023  18:31  <DIR>          obj
30/05/2023  23:10  <DIR>          Problemitas
01/06/2023  02:17          1.517 ProyectoC.cbp
01/06/2023  02:17          1.664 ProyectoC.depend
01/06/2023  02:29          1.333 ProyectoC.layout
01/06/2023  02:09           73 totales.out
```

### ➤ Como usar el programa

1) Para compilar y generar un ejecutable del código proporcionado debemos usar la siguiente sentencia

- **gcc -o [nombre\_del\_ejecutable] cuentaPalabras.c lista.c multiset.c lista.h multiset.h errores.h**
  - Donde **cuentaPalabras.c** contiene el main().
  - **lista.c** esta implementada con una simplemente enlazada sin nodos centinelas y sin un orden de inserción en la misma.
  - **multiset.c** esta implementada con un árbol trie.
  - **lista.h** contiene el encabezado de los métodos a implementar en **lista.c**
  - **multiset.h** contiene el encabezado de los métodos a implementar en **multiset.c**
  - **errores.h** contiene todos los errores contemplados en el diseño e implementación del código.

2) `[nombre_del_ejecutable] -h` ofrece un menú de opciones como el siguiente:



```
Seleccionar C:\Windows\system32\cmd.exe
C:\Users\Usuario\AppData\Local\Temp\cc7mjvSh.o: cuentaPalabras.c:(.text+0x7a6): undefined reference to `testArbolTrie'
collect2.exe: error: ld returned 1 exit status

C:\Users\Usuario\Desktop\ProyectoC>gcc -o ejecutable cuentaPalabras.c lista.c lista.h multiset.c multiset.h errores.h
C:\Users\Usuario\Desktop\ProyectoC>ejecutable -h
Contenido de los argumentos >>
Argumento 0: ejecutable
Argumento 1: -h

Cuenta las palabras en los archivos .txt de un directorio.
Uso: cuentapalabras [directorio de entrada]

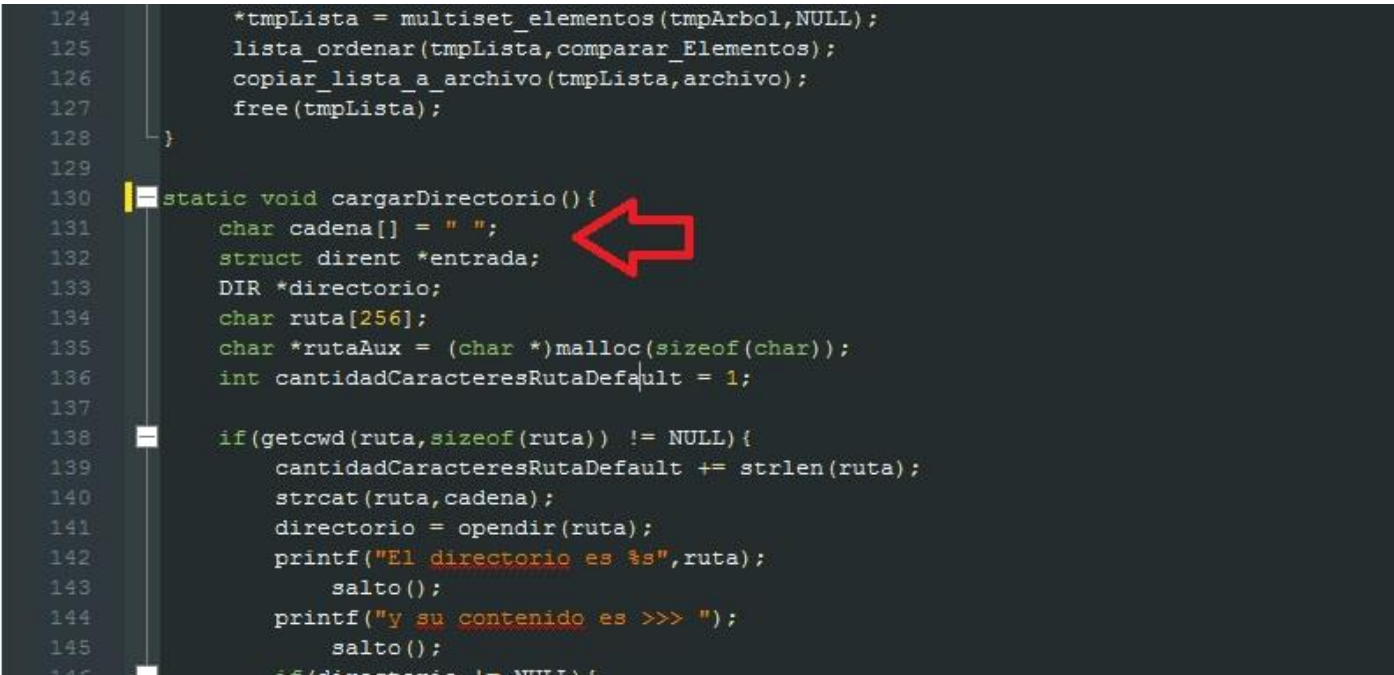
Opciones:
-h           Mostrar esta ayuda por pantalla.
-3           Test cuentaPalabras.
```

➤ Análisis de las opciones proporcionadas

- 1) Opción -1: Realiza un test de la implementación de la lista.
- 2) Opción -2: Realiza un test de la implementación multiset.
- 3) Opción -3: Ejecuta cuentaPalabras.

➤ Opción -3: cuentaPalabras

- 1) En caso de querer especificar un subdirectorio en específico deberán acceder a cuentaPalabras.c y modificar el siguiente apartado:



```
124     *tmpLista = multiset_elementos(tmpArbol, NULL);
125     lista_ordenar(tmpLista, comparar_Elementos);
126     copiar_lista_a_archivo(tmpLista, archivo);
127     free(tmpLista);
128 }
129
130 static void cargarDirectorio(){
131     char cadena[] = " ";
132     struct dirent *entrada;
133     DIR *directorio;
134     char ruta[256];
135     char *rutaAux = (char *)malloc(sizeof(char));
136     int cantidadCaracteresRutaDefault = 1;
137
138     if(getcwd(ruta, sizeof(ruta)) != NULL){
139         cantidadCaracteresRutaDefault += strlen(ruta);
140         strcat(ruta, cadena);
141         directorio = opendir(ruta);
142         printf("El directorio es %s", ruta);
143         salto();
144         printf("y su contenido es >>> ");
145         salto();
146         if(directorio != NULL){
```

➤ Observaciones importantes

- Sobre método el método **lista\_t multiset\_elementos(multiset\_t \*m, comparacion\_resultado\_t (\*f)(elemento\_t\*, elemento\_t\*));**
  - Por cuestiones de funcionalidad del proyecto y avalado por la catedra hemos modificado el parámetro **int** de **(\*f)(elemento\_t\*, elemento\_t\*))** por **comparación\_resultado\_t**