

Programación para Sistemas. EXAMEN PRÁCTICO PROGRAMACIÓN C
Curso 2019/2020 Enero 2020. Duración 1 hora y 30 minutos

Se desea implementar un programa en C con las siguientes funcionalidades:

- a) **(1.5 puntos)** El programa debe obtener un nombre de fichero por línea de comandos. Si al programa se le llama sin argumentos o con más de 1 argumento de entrada, el programa debe mostrar por la salida error el siguiente mensaje: *"Error: el programa espera un argumento de entrada correspondiente al fichero a tratar."* y finalizar su ejecución con código status 1.
- b) **(1.5 puntos)** El programa debe abrir en modo lectura el fichero correspondiente al argumento pasado en la línea de comandos. Si el fichero no existe o no es legible, el programa debe mostrar por la salida error el siguiente mensaje (que incluye el nombre del fichero que no se pudo abrir): *"Error: el fichero \"%s\" no existe o no es legible."* y finalizar su ejecución con código status 2.
- c) **(3 puntos)** El programa debe contar el número de líneas del fichero si se ha podido abrir y escribir en salida estándar el número de líneas del fichero. Para ello es necesario implementar y llamar a la función `contar_lineas` que cuenta las líneas del fichero pasado como argumento y devuelve el número de líneas leídas. Las líneas del fichero tienen un tamaño máximo de 80 caracteres.
- d) **(3 puntos)** Adicionalmente, la función `contar_lineas`, debe guardar en memoria la segunda línea del fichero (en caso de que exista), para lo cual habrá que implementar y llamar a la función `strdup2` (sin utilizar la librería de C `strdup`). La función recibe como argumento de entrada una cadena de caracteres y devuelve un puntero a la cadena de caracteres copiada (asignando memoria dinámica) si la operación ha tenido éxito, o bien devuelve NULL si ha habido error en la asignación de memoria.
Si se pudo realizar la copia correctamente, se debe escribir en salida estándar la línea copiada y liberar la memoria dinámica asignada.
- e) **(1 punto)** Al finalizar la ejecución el programa debe cerrar el fichero correspondiente si se pudo abrir y acabar con código status 0.

Rellenar la plantilla que se proporciona (sin usar hojas adicionales) con todo el código necesario para implementar los apartados pedidos, utilizando los huecos reservados.

Apellidos:

Nombre:

Matricula:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

```
/* FUNCIÓN APARTADO d) */
```

```
char *strdup2(const char *s)
{
    char *cadena;
```

```
/* CÓDIGO */
```

```
}
```

```
/* FUNCIÓN APARTADOS c) y d) */
```

```
/* Se recomienda usar char *fgets(char *str, int n, FILE *stream) */
```

```
int contar_lineas(FILE * file)
{
```

```
    char *segunda_linea;
```

```
/* DECLARACIÓN DE VARIABLES NECESARIAS */
```

```
/* CÓDIGO */
```

```
}
```

```
/* >>>>>>>>> CONTINUA POR ATRÁS >>>>>>>>> */
```

Apellidos:

Nombre:

Matricula:

```
int main(int argc, char *argv[])  
{
```

```
    /* DECLARACION DE VARIABLES NECESARIAS */
```

```
    /* CÓDIGO CORRESPONDIENTE AL APARTADO a) */
```

```
    /* CÓDIGO CORRESPONDIENTE AL APARTADO b) */
```

```
    /* CÓDIGO CORRESPONDIENTE AL APARTADO c) (llamada y salida) */
```

```
    /* CÓDIGO CORRESPONDIENTE AL APARTADO e) */
```

```
}
```