

Proyecto Semestre 1

LS Strategist



ÍNDEX

1 Enunciado	3
1.1 Objetivos	4
2 Funcionamiento	5
2.1 Opción 1: estrategia de neumáticos	6
2.2 Opción 2: código de radio	9
2.3 Opción 3: comunicación con el piloto	11
2.4 Opción 4: simulación de la carrera	20
2.5 Opción 5: salir	25
2.6 Opcionales	25
3 Consideraciones	26
4 Requisitos mínimos	27
5 Normativa general	28
6 Contenido de la memoria	29
7 Fechas de entrega	30

1 ENUNCIADO

El equipo de Fórmula 1 *Scuderia Ferrari* no está pasando por su mejor momento. No ganan un mundial de pilotos desde 2007 y el de constructores desde 2008. Estos resultados no están a la altura de un equipo de la talla de Ferrari. Tanto los directivos como los aficionados se están empezando a impacientar.

Los directivos han hecho una profunda reflexión y tras revisar situaciones como, por ejemplo, el Gran Premio de Abu Dhabi de 2010, donde un error de estrategia contribuyó a dejar escapar el título de pilotos de las manos de Fernando Alonso, han llegado a la conclusión de que un gran punto de potencial mejora es la gestión estratégica. Es por este motivo que desde la escudería han tomado la decisión de desarrollar un software que les ayude con la estrategia de los Grandes Premios.

Nos han escogido como desarrolladores de este programa, llamado LS Strategist, que ayudará con la elección de la estrategia de neumáticos, la comunicación con el piloto y la simulación de una hipotética carrera para probar que la estrategia sea la adecuada.

1.1 OBJETIVOS

Esta práctica ha estado diseñada con la intención de aplicar los siguientes conceptos y objetivos:

- Sentencias condicionales.
- Sentencias alternativas.
- Sentencias iterativas.
- Arrays y cadenas de caracteres.
- Ver las ventajas del uso de tipos propios.
- Entrar en contacto con la librería *math.h*.
- Aprender a compilar de forma manual un proyecto de C.

2 FUNCIONAMIENTO

Con tal de mantener una correcta apariencia y una usabilidad adecuada de nuestro programa, lo primero que deberemos mostrar cuando se ejecute LS Strategist será un pequeño mensaje de bienvenida acompañado de un menú con las distintas opciones que ofrece el programa. Las opciones son números enteros entre 1 y 5, ambos incluidos.

```
Bienvenido a LS Strategist!
```

```
Por favor , escoge una opcion del menu:
```

- ```
1. Estrategia de neumaticos.
2. Obtencion del codigo de radio .
3. Comunicarse con el piloto .
4. Simular carrera .
5. Salir .
```

```
Opcion seleccionada:
```

Output 1: Mensaje de bienvenida y menú inicial

Es importante tener claro que este menú deberá mostrarse siempre después de ejecutar cualquier opción hasta que no se escoja la opción 5, es decir, salir.

Además, como programadores se nos pide que nos aseguremos que en caso de que algún trabajador despistado introduzca un valor incorrecto el programa siga funcionando correctamente. En caso de que esto suceda, deberemos notificar, mediante un mensaje de error, la selección de una opción inválida y volver a pedir al usuario que indique qué opción desea seleccionar. Éste procedimiento se repetirá tantas veces como sea necesario hasta que el valor introducido sea válido.

```
Bienvenido a LS Strategist!
```

```
Por favor , escoge una opcion del menu:
```

- ```
1. Estrategia de neumaticos.  
2. Obtencion del codigo de radio .  
3. Comunicarse con el piloto .  
4. Simular carrera .  
5. Salir .
```

```
Opcion seleccionada: 27
```

```
ERROR: Los valores admitidos se encuentran entre el 1 y el 5.  
Por favor , vuelva a introducir la opcion seleccionada:
```

Output 2: Ejemplo de introducción de una opción incorrecta

2.1 OPCIÓN 1: ESTRATEGIA DE NEUMÁTICOS

La elección de los neumáticos adecuados y su correcta gestión es primordial en la Fórmula 1. Por este motivo se nos ha pedido que desarrollemos esta primera opción, que nos ayudará con la elección de éstos y el diseño de la estrategia de paradas.

Para comprender hasta qué punto tiene sentido la elección de los neumáticos, debemos saber que cuanto más blando sea un neumático, más rápido va a ir el monoplaza, pero a la vez durará menos y se tendrán que cambiar antes.

Para poder recomendar el neumático más adecuado a las necesidades de la carrera, el programa deberá formular una serie de preguntas.

En primer lugar se deberá preguntar al usuario si está lloviendo. Los únicos valores que admitiremos como respuesta son las letras s, S, n, N. Entendiendo que *s* o *S* equivalen a sí y *n* o *N* equivalen a no. Por lo tanto, será una respuesta **case insensitive**.

Posteriormente, deberemos preguntar qué nivel de degradación provoca la pista sobre los neumáticos, valor que nos indicarán con un número entero entre el 0 y el 4, ambos incluidos. El 0 significa una degradación mínima y el 4 la máxima. Una vez tengamos el índice de degradación, preguntaremos cuánto quieren priorizar el llevar un compuesto más rápido y, que en consecuencia, dure menos. Esto también nos lo indicarán mediante un entero , pero de 1 a 5, ambos incluidos, donde el 1 significa que no se prioriza la velocidad del neumático y el 5 significa que se prioriza al máximo.

Pongamos un ejemplo ilustrativo de la petición de datos:

```
Asistente de estrategia de neumaticos:
```

```
Esta lloviendo sobre la pista? (S/N): N
```

```
Que nivel de degradacion provoca la pista sobre el neumatico? (0-4): 2
```

```
Cuanto se quiere priorizar la velocidad del neumatico sobre la durabilidad?  
(1-5): 4
```

Output 3: Petición de datos de la opción 1

Cabe destacar que si alguno de los valores introducidos por el usuario no se encuentra dentro de los admitidos, deberá volverse a pedir el dato. Por ejemplo:

```
Asistente de estrategia de neumaticos:
```

```
Esta lloviendo sobre la pista? (S/N): N
```

```
Que nivel de degradacion provoca la pista sobre el neumatico? (0-4): 32
```

```
ERROR: Por favor, introduzca un valor entre 0 y 4
```

```
Que nivel de degradacion provoca la pista sobre el neumatico? (0-4): 2
```

```
Cuanto se quiere priorizar la velocidad del neumatico sobre la durabilidad?  
(1-5): 4
```

Output 4: Petición de datos incorrecta de la opción 1

Como se puede apreciar en el ejemplo, cuando se introduce un valor erróneo se deberá volver a pedir un nuevo valor y hasta que no se introduzca un valor válido no se pasará a la siguiente pregunta.

Una vez tengamos todos los datos a nuestra disposición deberemos sugerir el neumático y la estrategia de paradas más adecuada. Para hacer esto, tendremos en cuenta lo siguiente:

- Si **no está lloviendo**, el neumático saldrá de la **diferencia** entre el **índice de priorización de velocidad (IPV)** y el **índice de degradación (ID)**. Siempre utilizaremos el **valor absoluto**. Pongamos que tenemos un IPV de 5 y un ID de 1. Haciendo la diferencia de IPV - ID obtenemos un 4 y, por lo tanto, propondremos el compuesto **C4**. Si tenemos un IPV de 1 y un ID de 4 obtendremos un -3, que tiene un valor absoluto de 3, por lo que propondremos el **C3**. Si la diferencia entre ambos índices fuese de 2, propondríamos el compuesto **C2**, y así con todos los valores posibles.
- Si **está lloviendo**, el neumático saldrá también de la **diferencia** entre el **índice de priorización de velocidad** menos el **índice de degradación**, pero esta vez solo podremos proponer o el neumático **Intermedio (I)** o el neumático de **Full Wet (W)**. Si el resultado que obtenemos de la diferencia es mayor o igual a 3, propondremos el intermedio, mientras que si es menor, propondremos el de **Full Wet**. Igual que en el caso anterior, utilizaremos el **valor absoluto** de la diferencia.
- Para decidir la cantidad de paradas deberemos realizar, simplemente miraremos que **índice de priorización de velocidad** nos han introducido. Si éste es inferior a 3, indicaremos que se debe realizar una sola parada, si es igual a 3 indicaremos 2 paradas y si es mayor a 3, indicaremos 3.

Pongamos un ejemplo para clarificar las condiciones:

Asistente de estrategia de neumaticos:

Esta lloviendo sobre la pista? (S/N): N

Que nivel de degradacion provoca la pista sobre el neumatico? (0-4): 1

Cuanto se quiere priorizar la velocidad del neumatico sobre la durabilidad?
(1-5): 5

El compuesto adecuado es el C4 y deberan realizarse 3 paradas a boxes.

Output 5: Ejemplo de cálculo de estrategia en seco

Y si ha llovido:

Asistente de estrategia de neumaticos:

Esta lloviendo sobre la pista? (S/N): S

Que nivel de degradacion provoca la pista sobre el neumatico? (0-4): 1

Cuanto se quiere priorizar la velocidad del neumatico sobre la durabilidad?
(1-5): 3

El compuesto adecuado es el W y deberan realizarse 2 paradas a boxes.

Output 6: Ejemplo de cálculo de estrategia en mojado

2.2 OPCIÓN 2: CÓDIGO DE RADIO

Con tal de poder comunicarnos con el piloto mediante la radio deberemos generar un código de sintonización, el cual seguirá el siguiente formato: <LETRA><DIGITO_1><DIGITO_2>. Este código nos dará la clave que deberemos introducir con el fin de poder contactar con el piloto de una forma segura.

Para poder realizar el cálculo necesitaremos algunos datos que pediremos siguiendo la misma filosofía que en la opción 1. Pediremos:

- Número de dorsal del piloto (1-99)
- Número de la carrera (1-21)

Igual que en la opción 1, deberemos comprobar que los valores introducidos se encuentran dentro del rango adecuado y de no ser así, mostrar un mensaje de error y volver a pedirlos.

Generador de codigo de comunicacion:

Cual es el numero de dorsal del piloto con quien se quiere comunicar? 44

Cual es el numero de la carrera en el calendario? 0

ERROR: Has introducido un valor invalido

Cual es el numero de la carrera en el calendario?

Output 7: Ejemplo de introducción de datos en la opción 2

Para generar el código simplemente deberemos concatenar la letra del alfabeto correspondiente al número del dorsal del piloto (el 1 es A, el 2 es B, ... , y el 26 es la Z), seguido de dos dígitos. En caso de que el dorsal sea mayor a 26, volveremos a empezar el alfabeto. El dorsal 27 corresponderá a la A, el 28 a la B y así sucesivamente. Precisamente el segundo dígito del código corresponderá a las veces que se haya dado la vuelta al abecedario.

El primero corresponderá a la parte entera de la división del dorsal entre el número de carrera, es decir, si tenemos el dorsal 5 y estamos en la carrera número 2, el primer dígito del código será un 2 ($5/2 = 2,5$).

Pongamos un ejemplo:

Generador de codigo de comunicacion:

Cual es el numero de dorsal del piloto con quien se quiere comunicar? 44

Cual es el numero de la carrera en el calendario? 21

```
CODIGO GENERADO: R21
```

Output 8: Ejemplo de ejecución de la opción 2

En el ejemplo anterior, el piloto tiene el dorsal 44 y se encuentra en la carrera número 21 del campeonato. Por lo tanto, obtendremos el primer dígito (<DIGITO_1>) dividiendo, de forma entera, 44 entre 21, que da **2**. También tenemos que comprobar la posición de 44 en el alfabeto. Esto lo haremos dividiendo 44 entre el número de letras del abecedario (26). Ésta operación nos dará un cociente de **1** y un residuo de 18. El cociente representa el segundo dígito (<DIGITO_2>), es decir, el número de vueltas dado al abecedario, y el residuo será la letra correspondiente al dorsal (<LETRA>), en este caso la letra **R** (décimo octava letra del abecedario). Por lo tanto, el código generado es R21.

Si resultase que no se ha dado ninguna vuelta al abecedario, el segundo dígito deberá ser 0.

```
Generador de codigo de comunicacion:
```

```
Cual es el numero de dorsal del piloto con quien se quiere comunicar? 1
```

```
Cual es el numero de la carrera en el calendario? 1
```

```
CODIGO GENERADO: A10
```

Output 9: Ejemplo 2 de ejecución de la opción 2

2.3 OPCIÓN 3: COMUNICACIÓN CON EL PILOTO

Pese a esta gran ventaja competitiva respecto al resto de escuderías que supone el tenernos como programadores, la *Scuderia Ferrari* quiere aprovechar esta temporada para devolverle cierta espina clavada a la empresa americana llamada *Pord*, que les arrebató la victoria de las 24h de Lemans en 1966. Este año entrante, *Pord* creará su escudería para la F1, por lo que en Ferrari se han asegurado, mediante espionaje industrial, de no poder perder bajo ningún concepto.

Es por ello que han dado con el método ideal para la resolución de cualquier tipo de posible imprevisto durante la carrera, y es mediante la comunicación con el piloto. Nada más determinar su estrategia, Ferrari envía una cadena con su estrategia y la de todos los otros pilotos a su corredor para que los coches puedan indicarles si quien los adelanta supone un problema (tiene una estrategia mejor a la nuestra) o no (dada su estrategia, perderán solos).

Sin embargo, por ahora sólo tenemos que ser capaces de enviarle la información al piloto. Para ello, tendremos que utilizar el código de radio **que hayamos generado en el apartado anterior**. Así, lograremos cifrar la comunicación y evitar posible contraespionaje por parte de *Pord* o cualquier otro rival. Si nos equivocamos al introducir el código deberemos mostrar un error y volver a pedirlo hasta que **sea correcto o se introduzca la cadena "exit"**. Si aún no lo hemos generado, deberemos mostrar un mensaje de error y volver al menú.

Si el código es correcto, deberemos mostrar un mensaje indicando el dorsal del piloto con el que nos estamos comunicando y el número de la carrera actual.

```
Bienvenido a LS Strategist!
```

```
Por favor , escoge una opcion del menu:
```

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.
3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

```
Opcion seleccionada: 3
```

```
ERROR: Aun no se ha generado ningun codigo de cifrado para la radio.
```

```
Por favor , escoge una opcion del menu:
```

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.

3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

Opcion seleccionada:

Output 10: Ejemplo de introducción de una opción no completa

Bienvenido a LS Strategist!

[...]

Por favor, escoge una opcion del menu:

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.
3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

Opcion seleccionada: 3

Introduzca el codigo de radio: GH12

ERROR: Este codigo de radio no existe.

Introduzca el codigo de radio: LJ23

ERROR: Este codigo de radio no existe.

Introduzca el codigo de radio: exit

Por favor, escoge una opcion del menu:

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.
3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

Opcion seleccionada:

Output 11: Ejemplo de introducción de un código incorrecto

```
Bienvenido a LS Strategist!
```

```
[ ... ]
```

```
Por favor, escoge una opcion del menu:
```

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.
3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

```
Opcion seleccionada: 3
```

```
Introduzca el cogido de radio: R21
```

```
Codigo correcto.
```

```
Comunicandose con el piloto #44 en la carrera #21...
```

```
[ ... ]
```

Output 12: Ejemplo de introducción de un código correcto

Una vez hayamos introducido el código de radio, se nos desbloqueará la opción de enviarle las estrategias robadas al piloto en cuestión, que escribiremos nosotros mediante la siguiente estructura:

numCarrera|indiceDegradacion|llueve|numVueltas|numPilotos|<piloto_1>|...|<piloto_N>

Donde:

- **numCarrera:** Número de carrera en la que nos encontramos. Puede ser un número entre 1 y 21, ambos incluidos.
- **indiceDegradacion:** Índice con el que representamos el desgaste de los neumáticos. Puede ser un valor entre 0 y 4, ambos incluidos.
- **llueve:** Carácter **S o s** si va a llover y carácter **N o n** si no va a llover. Cualquier otro carácter será erróneo.
- **numVueltas:** Número de vueltas que habrá que dar al circuito. Tiene que ser un valor positivo, por lo tanto, **mayor** a 0, y no superior a 10.
- **numPilotos:** Cantidad de pilotos que ha logrado clasificarse para la carrera. Puede haber como máximo 20 pilotos y como mínimo 2.

Cada uno de los piloto, tantos como indique el campo *numPilotos*, seguirá el siguiente formato:

**dorsal-nombre-fechaNacimiento-compuesto-numParadas-
numVueltaParada1/numVueltaParada2/.../numVueltaParadaN**

Donde:

- **dorsal:** Dorsal del piloto en cuestión. Puede tener un valor entre 1 y 99, ambos incluidos.
- **nombre:** Nombre del piloto. Hay que tener en cuenta que **puede contener espacios** y nunca superará los 25 caracteres.
- **fechaNacimiento:** Fecha de nacimiento del piloto. Seguirá el formato "dd/mm/aaaa". Esto quiere decir que tanto el día como el mes tendrán dos caracteres **siempre**. También cabe mencionar que habrá que corroborar que la fecha sea correcta, es decir, los días estén entre 1 y 31 (obviando febrero), los meses entre 1 y 12, y los años bisiestos sean correctos.
- **compuesto:** Neumáticos con los que saldrá a correr el piloto. Se asegura que **siempre será un compuesto válido**, por lo que no será necesario comprobarlo.
- **numParadas:** Número de paradas que va a realizar el piloto en cuestión. Tiene que ser un valor positivo, por lo tanto, **mayor** a 0, e inferior a 11.
- **numVueltaParadas:** Número de vuelta en las que sucederá cada parada, separadas por un / entre ellas. Tiene que haber tantas como las indicadas en el valor anterior (numParadas), por lo que habrá que comprobar que coincida. Si no coincide, se considerará que es un valor erróneo.

Además:

- "|": Carácter que separa bloques de información.
- "-": Carácter que separa información de un mismo piloto.
- "/": Carácter que separa las diferentes paradas de un piloto.

Es importante recalcar que en caso de que haya cualquier valor erróneo en la cadena, ésta se considerará inválida y se volverá al menú inicial.

De la misma manera, cabe decir que **el orden en el que introducimos a los pilotos es el orden de salida de la carrera**, comenzando desde la primera línea hasta la última.

Así pues, un ejemplo válido de cadena sería:

**1|2|n|10|2|1-Kasill Ego-29/02/2000-C4-2-4/8|40-Siempre VanPrimero-
01/012/1997-C4-1-6**

Nota: Tanto en este como en el resto de ejemplos que os damos, no mostramos el contenido en una sola línea por vuestra comodidad, pero cuando estemos dentro del programa sí que deberá ser una línea.

Una vez hayamos introducido toda la información y hayamos comprobado que es correcta, deberemos notificar de vuelta en boxes que hemos logrado transferir la información correctamente al piloto.

Nota: Cuando estemos leyendo la cadena de información, al encontrar el **primer** error deberemos de dejar de leer y volver al menú principal. Por lo tanto, aunque en una cadena haya más de un error, únicamente informaremos del primero.

```
Bienvenido a LS Strategist!

[ ... ]

Por favor , escoge una opcion del menu:

    1. Estrategia de neumaticos.
    2. Obtencion del codigo de radio.
    3. Comunicarse con el piloto.
    4. Simular carrera.
    5. Salir.

Opcion seleccionada: 3

Introduzca el cogido de radio: All

Codigo valido!
Comunicandose con el piloto #3 en la carrera #21...

Introduzca la informacion:
21|2|n|10|2|3 - Kasill Ego-29/02/2000-C4-2-4/8|
40-Siempre VanPrimero-01/12/1997-C4-1-6

Informacion validada y enviada correctamente!
```

Output 13: Ejemplo de introducción de una informaicón correcta

Sin embargo, cuando encontremos cualquiera de los posibles errores que se pueden dar en nuestra cadena, deberemos mostrar un mensaje de error informativo y volver a pedir la información **hasta que sea correcta o nos entren la cadena "exit"**.

```
Bienvenido a LS Strategist!
```

```
[ ... ]
```

Por favor , escoge una opcion del menu:

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.
3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

Opcion seleccionada: 3

Introduzca el cogido de radio: A11

Codigo valido!

Comunicandose con el piloto #3 en la carrera #21...

Introduzca la informacion:

210|2|n|10|2|3 - Kasill Ego-20/02/2000-C4-2-4/8|
40-Siempre VanPrimero-01/12/1997-C4-1-6

ERROR: El numero de carrera es incorrecto!

Introduzca la informacion:

Output 14: Ejemplo de introducción de un número de carrera incorrecto

Bienvenido a LS Strategist!

```
[ ... ]
```

Por favor , escoge una opcion del menu:

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.
3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

Opcion seleccionada: 3


```
Introduzca el cogido de radio: A11

Codigo valido!
Comunicandose con el piloto #3 en la carrera #21...

Introduzca la informacion:
21|2|n|10|2|3 - Kasill Ego-30/02/2000-C4-2-4/8|
40-Siempre VanPrimero-01/12/1997-C4-1-6

ERROR: La fecha de nacimiento del piloto Kasill Ego es incorrecta!

Introduzca la informacion:
```

Output 15: Ejemplo de introducción de una fecha incorrecta

```
Bienvenido a LS Strategist!

[ ... ]

Por favor , escoge una opcion del menu:

    1. Estrategia de neumaticos.
    2. Obtencion del codigo de radio.
    3. Comunicarse con el piloto.
    4. Simular carrera.
    5. Salir.

Opcion seleccionada: 3

Introduzca el cogido de radio: A11

Codigo valido!
Comunicandose con el piloto #3 en la carrera #21...

Introduzca la informacion:
21|2|n|10|2|3 - Kasill Ego-20/02/2000-C4-2-4/8|
40-Siempre VanPrimero-01/12/1997-C4-1-2/4/8

ERROR: El piloto Siempre VanPrimero tiene 1 parada, pero ha indicado 3!

Introduzca la informacion:
```

Output 16: Ejemplo de no concordancia entre numParadas y las numVueltaParadas

```
Bienvenido a LS Strategist!
```

```
[ ... ]
```

```
Por favor , escoge una opcion del menu:
```

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.
3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

```
Opcion seleccionada: 3
```

```
Introduzca el cogido de radio: A11
```

```
Codigo valido!
```

```
Comunicandose con el piloto #3 en la carrera #21...
```

```
Introduzca la informacion:
```

```
21|2|n|10|0|
```

```
ERROR: Tiene que haber mas de 1 piloto
```

```
Introduzca la informacion:
```

Output 17: Ejemplo de número de pilotos incorrecto

```
Bienvenido a LS Strategist!
```

```
[ ... ]
```

```
Por favor , escoge una opcion del menu:
```

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.
3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

```
Opcion seleccionada: 3

Introduzca el cogido de radio: A11

Codigo valido!
Comunicandose con el piloto #3 en la carrera #21...

Introduzca la informacion:exit

Cancelando envio...

Bienvenido a LS Strategist!

Por favor , escoge una opcion del menu:

    1. Estrategia de neumaticos.
    2. Obtencion del codigo de radio.
    3. Comunicarse con el piloto.
    4. Simular carrera.
    5. Salir.

Opcion seleccionada:
```

Output 18: Ejemplo de cómo salir sin introducir datos correctos

2.4 OPCIÓN 4: SIMULACIÓN DE LA CARRERA

Una vez hayamos logrado enviar nuestra información al piloto, sólo queda la posibilidad de esperar a los resultados de nuestro esfuerzo. Y como dice el refrán "el que espera, desespera", hemos decidido añadir una última funcionalidad que nos permita simular la carrera.

Para ello, deberemos haber realizado previamente las opciones 1, 2 y 3 del menú, por lo que deberemos controlarlo.

```
Bienvenido a LS Strategist!
```

```
Por favor, escoge una opcion del menu:
```

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.
3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

```
Opcion seleccionada: 4
```

```
ERROR: Aun no has configurado la estrategia de neumaticos
```

```
Por favor, escoge una opcion del menu:
```

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.
3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

```
Opcion seleccionada:
```

Output 19: Ejemplo de simulación de carrera con error por la no ejecución de la opción 1

```
Bienvenido a LS Strategist!
```

```
Por favor, escoge una opcion del menu:
```

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.

3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

Opcion seleccionada: 4

ERROR: Aun no has generado el codigo de radio.

Por favor, escoge una opcion del menu:

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.
3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

Opcion seleccionada:

Output 20: Ejemplo de simulación de carrera con error por la no ejecución de la opción 2

Bienvenido a LS Strategist!

Por favor, escoge una opcion del menu:

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.
3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

Opcion seleccionada: 4

ERROR: Aun no has enviado ninguna informacion al piloto.

Por favor, escoge una opcion del menu:

1. Estrategia de neumaticos.
2. Obtencion del codigo de radio.
3. Comunicarse con el piloto.
4. Simular carrera.
5. Salir.

Opcion seleccionada :

Output 21: Ejemplo de simulación de carrera con error por la no ejecución de la opción 3

Si por el contrario ya hemos provisto al piloto con la información necesaria, podemos pasar a simular la carrera. Para ello, hemos desarrollado una función matemática que representa más o menos las posibilidades de ganar de cada uno de los pilotos:

$$posibilidades = 100 - \sqrt{|A^2 + B^2|} - |\cos(((C/D) * 100 * 2/3) + (1/3 * E)) * 40|$$

Donde:

- **A:** Diferencia entre el neumático recomendado según lo que queríamos (opción 1) y el que ha puesto el piloto (opción 3). Si el neumático es el mismo, a tendrá un valor de 0.
Los tenemos categorizados de la siguiente manera:
Si nos proponen CX (siendo X cualquier valor) y nosotros usamos intermedio (I), asignaremos a A un valor de 1, mientras que si usamos Full Wet (W) A valdrá 2.
Si nos proponen el intermedio (I) y hacemos uso de CX o de W, el valor de A es de 1.
Si nos proponen Full Wet (W), si hacemos uso de CX el valor de A será de 2, mientras que si nos hemos decantado por intermedio será de 1.
- **B:** Diferencia entre cantidad de paradas recomendadas (opción 1) y las que va a hacer el piloto (opción 3).
- **C:** Cantidad de pilotos que **salen** por delante del piloto en cuestión.
- **D:** Cantidad de pilotos totales en la carrera.
- **E:** Factor "suerte" aleatorio que siempre irá comprendido entre 0 y 7, ambos incluidos.

Nota: El primer factor que resta en la fórmula es porque nuestro programa es perfecto y todo aquello que no sea ajustarse a nuestras recomendaciones es un error. El segundo, es porque con *Pord* en la lucha por el título, habrá mucha más siniestralidad en los GP, por lo que cuanto más atrás estés en la salida más posibilidades tienes de encontrarte un accidente, ergo menos tienes de ganar.

Como podréis suponer, este cálculo deberá de realizarse para todos los pilotos de la carrera, incluido el nuestro.

Con todo esto, también cabe recalcar que una vez entremos en esta opción nosotros no debemos de introducir nada, nuestro programa ya dispondrá de toda la información necesaria porque se la habremos facilitado en los apartados anteriores.

Después de mostrar la clasificación en orden de salida, es decir, no hace falta ordenarlos. También se deberá entregar el trofeo al campeón.

- 4. Simular carrera .
- 5. Salir .

Opcion seleccionada:

Output 22: Ejemplo de ejecución de la cuarta opción del menú

Nota: Los valores porcentuales mostrados en este ejemplo no se corresponden a los que puede dar el cálculo si ponemos la cadena de ejemplo de más arriba, son únicamente orientativos.

2.5 OPCIÓN 5: SALIR

Cuando el usuario seleccione esta opción, se deberá finalizar la ejecución del programa, dando a nuestro equipo de F1 un mensaje de despedida.

```
Bienvenido a LS Strategist!  
  
Por favor, escoge una opcion del menu:  
  
1. Estrategia de neumaticos.  
2. Obtencion del codigo de radio.  
3. Comunicarse con el piloto.  
4. Simular carrera.  
5. Salir.  
  
Opcion seleccionada: 5  
  
Gracias por usar el programa!  
Suerte en la siguiente carrera.
```

Output 23: Ejemplo de introducción de la opción salir

2.6 OPCIONALES

De forma opcional, podéis utilizar funciones y procedimientos para estructurar el código. Recordad que hay que seguir el estilo especificado en la guía de estilos.

Como bien dice su nombre, esto es opcional, por lo que no se penalizará su ausencia. En cambio, si se implementa sí que repercutirá de forma positiva en la nota del proyecto.

3 CONSIDERACIONES

Para la implementación de esta práctica tened en cuenta las siguientes consideraciones:

1. Se debe seguir el formato mostrado en los ejemplos.
2. Cuando el valor a introducir sea un número entero, se os garantiza que siempre se introducirá un entero, nunca un carácter o cadena de caracteres.
3. El valor seleccionado para las opciones del menú tiene que ser un **entero** entre 1 y 5, ambos incluidos. Cualquier otro valor será considerado erróneo.
4. Para indicar si va a llover o no se introducirá el **carácter S/s o N/n**, respectivamente. Cualquier otro carácter será considerado erróneo.
5. El nivel de degradación de los neumáticos tiene que ser un valor **entero** entre 0 y 4, ambos incluidos.
6. El nivel de priorización de la velocidad tiene que ser un **entero** entre 1 y 5, ambos incluidos.
7. El dorsal de los pilotos tiene que ser un valor **entero** entre 1 y 99, ambos incluidos.
8. El número de carrera tiene que ser un **entero** entre 1 y 21, ambos incluidos.
9. El nombre del piloto será una **cadena de caracteres** de como máximo **25** caracteres. Recordad que también se necesita espacio para guardar el carácter \0.
10. La fecha **siempre** sigue el formato dd/mm/aaaa. Tiene que ser una fecha válida.
11. El número de paradas que va a hacer el piloto tiene que ser un **entero** entre 1 y 10, ambos incluidos.
12. El número de paradas especificado tiene que coincidir con las paradas que se van a hacer.
13. El número de vueltas tiene que ser un **entero** superior a 0.
14. El número de pilotos tiene que ser un **entero** entre 2 y 20, ambos incluidos.
15. El factor aleatorio tiene que ser un **entero** entre 0 y 7, ambos incluidos.
16. En el código del programa no puede escribirse la posición o valor numérico de ningún carácter de la tabla ASCII.

4 REQUISITOS MÍNIMOS

Esta práctica debe cumplir con una serie de requisitos mínimos de calidad para poder evaluarla:

- El código debe estar correctamente comentado de manera que sea legible sin dificultades.
- Se debe seguir la guía de estilos de la asignatura.
- El programa debe desarrollarse íntegramente en el entorno Linux de Matagalls, usando el editor vim para escribir el código C y el compilador gcc para la obtención del ejecutable correspondiente.
- No se puede usar ninguna herramienta ni instrucción del lenguaje C que no se haya tratado en clase
- No se aceptará ninguna práctica que no cumpla la normativa de prácticas.

5 NORMATIVA GENERAL

El proyecto del 1º semestre consta de diversas funcionalidades que se evaluarán como una única práctica. Todas las funcionalidades tienen que estar implementadas correctamente. En el supuesto de no entregar el proyecto, la nota de la práctica en el sistema de información debe constar como NP (No Presentado). En caso de que alguno de los elementos a evaluar (calidad de código, calidad de ejecución, memoria o entrevista) no esté aprobado, la nota de la práctica en el sistema de información debe constar como la nota más baja entre los diferentes apartados.

Del proyecto se evaluará por separado la calidad de ejecución y la calidad de código. La nota de la memoria será común para ambos y la nota de entrevista individual para cada miembro del grupo.

Para poder evaluar el proyecto, éste tiene que seguir la nomenclatura y formato especificados en el apartado 7.

Nota Proyecto 1º Semestre = Calidad Ejecución (40 %) +
Calidad Código (20 %) + Memoria (30 %) + Entrevista (10 %)

6 CONTENIDO DE LA MEMORIA

La memoria debe contener **OBLIGATORIAMENTE** los siguientes apartados:

- Portada de la practica rellena.
- Índice numerado.
- Breve resumen del enunciado del proyecto y cómo se ha decidido diseñar la práctica (en caso de utilizar tipos propios y/o funciones, también hay que explicarlos).
- Diagrama de actividad de la opción 3 (Comunicación con el piloto)
- Responder las siguientes preguntas:
 1. ¿Cómo harías un bloque de código que se comportase como un *if-else* pero sin usar el *else*? Pon un ejemplo.
 2. ¿Qué significa *case sensitive*? ¿Y *case insensitive*?
 3. ¿Qué es y qué hace el *define*?
 4. ¿Por qué hay que utilizar la función *srand* antes de poder usar *rand*?
 5. ¿Por qué hay que limpiar el *buffer* después de un *scanf*?
- Problemas observados y cómo se han solucionado.
- Tiempo de realización de la práctica con una gráfica incluida y una explicación de ésta (Fases y Memoria).
- Contenido extra que se haya podido añadir. Este apartado es obligatorio para aquellos que hayan puesto algo de contenido extra a la práctica que no esté en este documento.
- Conclusiones elaboradas.

Cualquier memoria que no cumpla el formato será devuelta y la práctica será considerada no apta para aprobarla hasta que el alumno no la entregue correctamente.

7 FECHAS DE ENTREGA

Esta práctica contiene diferentes *checkpoints* que son opcionales pero recomendables para el correcto progreso de la práctica. A diferencia de una pre-entrega, los *checkpoints* no tienen nota. Simplemente se revisará tanto el código como la ejecución y se darán recomendaciones para poder mejorar la implementación. Este proceso se hará de forma **presencial** en los horarios de dudas de los becarios. Todos aquellos que quieran optar a hacer esta revisión deberán entregar antes del cierre del pozo correspondiente al *checkpoint*. Las fechas serán las siguientes:

- *Checkpoint* 1: opción 1, 2 i 3 (01/12/2019).
- Entrega sobre 10 (15/12/2019). Entrevistas 19/12/2019 y 20/12/2019.
- Entrega sobre 8 (07/06/2020). Entrevistas del 10/06/2020 al 12/06/2020.
- Entrega sobre 6 (05/07/2020). Entrevistas del 13/07/2020 al 15/07/2020.

El formato de entrega para los *checkpoints* deberá ser el siguiente:

Archivo comprimido con el nombre **PJ1_<login1>_<login2>.tar.gz** y contendrá el código.

Cualquier entrega que no cumpla con los objetivos de cada *checkpoint* no será aceptada.

El formato de entrega final deberá ser el siguiente:

Archivo comprimido con el nombre **PJ1_<login1>_<login2>.tar.gz** y contendrá:

- PJ1_<login1>_<login2>.c
- Memoria en PDF

Cualquier entrega fuera de plazo o por correo no será aceptado. Esto se aplica tanto a los *checkpoints* como a las entregas finales.