



## Prueba tipo examen (sin valor en la nota final).

**Normativa:** El examen se debe resolver con los conceptos vistos en clase: funciones, procedimientos, for, while, listas, cadenas, tuplas...

**Criterio de evaluación:** Cada apartado en el que aparezca la puntuación se evaluará de la siguiente manera:

60% de la puntuación si funciona.

30% de la puntuación si se usan las estructuras de datos adecuadas y éstas se utilizan adecuadamente.

10% de la puntuación calidad del código: nombrado de variables y estructuras de datos, comentarios en el código, etc.

## Playlist IES FBMOLL

1. Deseamos realizar un programa que nos ayude a gestionar nuestras playlist con las canciones e información relacionada sobre éstas. **(programa principal 1 punto)**. (Se asume que durante la ejecución el usuario introducirá datos adecuados, por tanto, no nos debemos preocupar por si introducen string en una petición por teclado de un flotante, etc). Con la información que hay que almacenar, y que a continuación se detalla, **selecciona la estructura de datos adecuada**.

**CONSEJOS:** Realizar primero la opción la versión 1, y si quedan fuerzas versión 2.

**Versión1:** Implementar las opciones de playlist, es decir, opción 1,2.

**Versión 2:** Implementar las opciones de canciones y listar playlist, es decir, opción 3,4 y 5.

Inicialmente no hay nada almacenado, por lo que el programa debe dar solución a añadir/borrar tantas playlist como queramos, y añadir/borrar tantas canciones como queramos. Además al finalizar cualquier operación (excepto Salir), **devolverá al usuario al menú principal**. Para ello debe realizar lo siguiente:

- a) **Mostrar un menú** por pantalla para seleccionar cada una de las operaciones que puede realizar el usuario. Al finalizar cada una de ellas, el programa pedirá de nuevo si quiere realizar alguna otra operación. Para simplificar daremos una opción de salir y finalizará el programa: **(1 punto)**

```

=====
= Playlist IES FBMOLL          =
=====
= 1) Añadir playlist           =
= 2) Borrar playlist           =
= 3) Añadir canciones a playlist =
= 4) Buscar canción por título =
= 5) Listar playlist           =
= 6) Salir                     =
=====
Selecciona la opción que desea:

```

- b) **Función “Añadir playlist”**. Cuando el usuario seleccione esta opción el programa pedirá el nombre de la playlist, que será por el que se identifique la playlist. Para simplificar, este nombre se almacenará siempre en mayúsculas. Además pedirá la descripción de la play list y guardaremos la fecha de creación que será la fecha actual (función time). Además, antes de insertar debe verificar que actualmente no existe ninguna play list con el mismo nombre, de ser así, enviará un mensaje informativo, y devolverá al usuario al menú principal sin insertar la playlist. **(1 punto)**
- c) **Función “Borrar playlist”**. Pedirá al usuario el nombre de la play list, si la encuentra la borrará, si no la encuentra le informará al usuario de que no la ha encontrado. **(1 punto)**
- d) **Función “Añadir canciones a la playlist”**. Pedirá al usuario el nombre de la playlist, si no la encuentra le informará al usuario y volverá al menú principal. Si la encuentra pedirá la información de la canción: el nombre de la canción, que será la que identifique la canción, el título lo almacenaremos en mayúsculas. Además antes de insertar, verificaremos que la canción no está ya en la playlist, de lo contrario nos informará, y nos volverá a pedir otra canción o la opción de salir de añadir canciones. También guardaremos la descripción de la canción, el año de la canción y la duración. **(2 puntos)**
- e) **Función “Buscar canción”**. Pedirá al usuario un título de una canción, y la buscará en cualquier playlist, si la encuentra nos informará de la playlist en la que se encuentra con el nombre de la playlist, descripción y fecha de creación. Además nos dará toda la información de la canción que tenemos guardada (título, descripción...). Si no la encuentra informará al usuario que no la ha encontrado. **(2 puntos)**
- f) **Procedimiento “Listar playlist”**. Pedirá al usuario una play list, y mostrará una cabecera con la información de la playlist, y a continuación toda la información de las canciones de la playlist. Si no encuentra la playlist, informará de que no la ha encontrado. **(2 puntos)**