

**Apellido, Nombre:**

**Padrón:**

## **Fundamentos de Programación**

### **Exámen final - 19/12/2024**

**Condición de aprobación:** Tener 2 ejercicios **Bien**.

#### **Aclaraciones:**

- Un ejercicio práctico se considera **Bien** cuando además de cumplir con lo pedido, aplica las buenas prácticas profesadas por la cátedra.
- En un ejercicio la parte práctica equivale al 75% de la calificación de ese punto mientras que la parte teórica equivale al 25%.

### **Ejercicio 1**

#### **Parte práctica**

Merida participó en un torneo de arco y flecha con otros 4 participantes, y quiere saber quién ganó.

Se tiene un archivo con los IDs y nombres de cada participante, y otro con los IDs y puntos que sacó cada participante en diferentes tiros que realizó.

Los formatos de los archivos son los siguientes:

#### **participantes.csv:**

ID;NOMBRE

#### **puntos.csv**

ID;CANT\_PUNTOS

#### **Se pide**

Dado ambos archivos mencionados, crear una función que devuelva el **nombre** del ganador.

## Parte teórica

1. Si tengo los siguientes tipos de datos:

```
typedef struct ingrediente {
    char tipo; // Salado (S), Dulce (D), Ácido (A)
    char nombre[MAX_NOMBRE];
} ingrediente_t;

typedef struct plato {
    char nombre[MAX_NOMBRE];
    ingrediente_t ingredientes[MAX_INGREDIENTES];
    bool disponible;
} plato_t;

typedef struct menu {
    bool actualizado;
    plato_t platos[MAX_PLATOS];
} biblioteca_t;
```

Especificar si el acceso que se está realizando es correcto, y en caso de serlo, especificar el tipo de dato al que se está accediendo:

- a. menu.platos[0].ingredientes
- b. menu.platos.ingredientes[0].tipo
- c. &(menu.platos[0].ingredientes[0])
- d. menu.platos[0].ingredientes[0].nombre

## Ejercicio 2

### Parte práctica

Alex y Justin Russo (Los Hechiceros de Weberly Place) encontraron un libro de hechizos nuevos, donde había uno para crear portales. Abrieron portales y dentro de ellos abrían otros. Practicaron durante tanto tiempo que se perdieron.

Ayudalos a encontrar su camino a "casa", sabiendo que **si recorren 20 portales se quedan sin magia**, y no van a poder volver.

```
typedef struct portal{
    int col_siguiete;
    int fila_siguiete;
    bool es_seguro;
    char nombre_lugar[MAX_LUGARES]; // "casa", "selva", "supermercado"
} portal_t;
```

### Se pide

Crear una función **recursiva** que reciba una matriz de portal\_t, la cual representa los portales creados por los hermanos, y devuelva si pudieron volver a "casa" o no.

### Aclaraciones

- Se empieza a recorrer desde la posición (0,0) de la matriz.

### Parte teórica

1. ¿Cuáles son los elementos de una función recursiva?
2. ¿Qué diferencias hay a la hora de trabajar con un vector o con un archivo?

## Ejercicio 3

### Parte práctica

La fraternidad de Mike y Sully va a hacer una fiesta, necesitan armar la playlist. Para eso, todos los integrantes pusieron sus temas favoritos en un archivo, pero algunos tienen mal gusto musical, así que hay que filtrar esa lista para que sea cool.

Dado un archivo de canciones con el siguiente formato:

```
TITULO_CANCION;GENERO;DURACION;ARTISTA;CALIFICACION_SPOTIFY
```

### Se pide

Crear un archivo con las canciones que sean cool. Una canción es cool cuando cumple con todas las siguientes condiciones:

- El título no contiene la palabra **"triste"**.
- La duración es de menos de 3 minutos.
- Es de pop o de rock.
- Tiene calificación más de 7 en Spotify.

Además, una canción siempre es cool cuando la canta **María Becerra**, sin importar los demás atributos.

### Aclaración

- La duración y la calificación son números enteros.

### Parte teórica

1. ¿Qué campos internos debería tener un `struct vector_dinamico_t`? ¿Y un `struct lista_enlazada_t`?
2. ¿Qué uso tiene cada campo mencionado en el punto anterior?