



Apellido, Nombre:

Mail:

Padrón:

Teórico / Práctico - Entrego hojas

1:

2:

3:

4:

5:

Nota Final

.....

Aclaraciones:

- Antes de comenzar a resolver el parcial, complete sus datos en esta hoja, y al finalizarlo, firme todas las hojas.
- Los ejercicios deben ser implementados en el lenguaje de programación C, respetando las buenas prácticas.
- Para cada ejercicio práctico se recomienda realizar un análisis y un diagrama del problema y la solución.
- Se deben numerar TODAS las hojas e inicializarlas con nombre, apellido, padrón y cualquier otra información que considere necesaria.
- La aprobación del parcial está sujeta a la aprobación de al menos el 60% del mismo.

Ejercicios:

Fluffy, es un perro de 3 cabezas muy apreciado por Hagrid, desde cachorro y para que no se convierta en una bestia, lo hace jugar un rato por día.

Dada la voracidad y fuerza de Fluffy, los juguetes no tienen muuuucha vida útil...

```
typedef struct juguete {
    char nombre[MAX_NOMBRE];
    int estado; // del 1 al 10 siendo 1 pesimo y 10 excelente
    int cabeza_ultimo_uso; // 1, 2 o 3
    int usos;
} juguete_t;
```

Se tiene 1 vector de juguetes:

```
juguete_t juguetes[MAX_JUGUETES];
```

1. Ron ordenó la huerta descendientemente por densidad de calabazas (los surcos más densos, es decir los que tienen más calabazas por metro están al principio).
La densidad está dada por cantidad de calabazas dividido la longitud del surco.

- a. Cree un algoritmo que ordene descendientemente los juguetes por cantidad de usos.
- b. ¿Puede utilizar la búsqueda binaria en el vector obtenido en el punto a) para buscar por nombre? ¿Por qué?

2. Al momento de ordenar la habitación y guardar los juguetes Hagrid le pareció ver que el "Chanco Chifle" no da para más.

- a. Cree un algoritmo que dado un vector ordenado por nombre del juguete elimine al "Chanco Chifle" solo si su estado es 1 (es decir, está en las últimas), manteniendo el orden del vector.



"Todo hay que volver a inventarlo..." - J.C.

3. Se tiene el siguiente algoritmo recursivo que muestra los nombres de los juguetes, el vector está ordenado alfabéticamente por nombre.

```
void mostrar_nombres(juguete_t juguetes[MAX_JUGUETES], int tope, int i){  
    mostrar_nombres(juguetes, tope, i+1);  
    printf("%s\n", juguetes[i]);  
}
```

Hagrid, en reportaje con el periódico local, declaró: "Me muestra los juguetes ordenados por nombre, pero al revés de cómo los quiero, yo que los muestre con el orden que tiene el vector", y con cara de confundido agregó "Además, me sale un mensaje que dice... -Segmentation fault- no entiendo nada!".

- a. ¿Cuáles son las fallas que encuentra en el algoritmo recursivo?
 - b. Arréglo para que funcione sin fallar y cómo quiere Hagrid.
4. En una de sus aburridas tardes, Hagrid quiere saber cual es el estado general de todos los juguetes, para eso, necesita que le creemos un vector que:
- En la posición 0 tendrá la cantidad de juguetes que tienen 0 usos.
 - En la posición 1 tendrá la cantidad de juguetes que tienen 1 uso.
 - En la posición 2 tendrá la cantidad de juguetes que tienen 2 usos.
 - ...

Es importante recorrer el vector de juguetes una sola vez.

5. Antes de quedar bajo la tutela de Hagrid, Fluffy vivía muy contento en el bosque, caminaba, caminaba sin cesar...
- Un día, cuando aún no sabía bien cómo manejarse, se perdió en una parcela del bosque. Una de sus cabezas dijo, 'Muchachas, no perdamos la calma, saldremos de ésta... O no...'. Imaginando que el bosque es como una matriz de caracteres, que se reciben las coordenadas donde está Fluffy, que los árboles están representados por 'A'.
- a. Crear una función que determine si Fluffy puede salir en línea recta, horizontal o verticalmente, de la parcela en la que se encuentra, sin cruzarse con un árbol.