

Apellido, Nombre:

Padrón:

Fundamentos de Programación

Exámen parcial - 16/10/2025

Tema El Barto

Condición de aprobación: Se debe tener al menos 1 (un) ejercicio con Bien en cada parte (teórica y práctica) y adicionalmente otro ejercicio con Regular o más. Para que un ejercicio práctico esté Bien tiene que cumplir con las buenas prácticas de programación establecidas por la cátedra.

Parte teórica

Ejercicio 1

a. Se tiene un programa que ordena un vector. Podemos ver el estado del vector en cada paso:

6	1	7	2	4	3	5
---	---	---	---	---	---	---

6	1	5	2	4	3	7
---	---	---	---	---	---	---

3	1	5	2	4	6	7
---	---	---	---	---	---	---

?	?	?	?	?	?	?
---	---	---	---	---	---	---

?	?	?	?	?	?	?
---	---	---	---	---	---	---

2	1	3	4	5	6	7
---	---	---	---	---	---	---

1	2	3	4	5	6	7
---	---	---	---	---	---	---

¿Cuales son los dos estados faltantes del vector? ¿Qué método de ordenamiento se está usando? ¿Qué esperarías ver si el método fuera otro?

b. Explicar paso a paso como usarías búsqueda binaria para encontrar el número 6 en el vector final.

Ejercicio 2

Dado el siguiente fragmento de código (se puede asumir que compila correctamente):

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

int main(){
    char* nombre = malloc(sizeof(char)*6);
    int* edad = 43;
    strcpy(nombre, "Homero");

    printf("%i", strlen(nombre) + *edad);
}
```

a. Enumerar todos los errores de memoria que se hacen en el programa. Para cada uno, explicar porque es un error y como se debería arreglar.

b. ¿A qué nos referimos cuando hablamos de aritmética de punteros?

Parte práctica

Ejercicio 1

A Bart le aburre ir a la escuela, por lo que siempre trata de buscar una razón por la cual burlarse de los demás.

Esta vez, en el colectivo, decide espiar todas las loncheras de sus compañeros para saber cuantos vegetarianos hay que llevaron ensalada para comer y cantarles **no vives de ensalada**. Sin embargo, si esa persona llevó papas fritas, no se va a burlar porque quiere que le comparta.

Teniendo el struct *lonchera_t*:

```
typedef struct lonchera {
    // "ensalada", "papas fritas", "hamburguesa"
    char alimentos[MAX_LETRAS][MAX_ALIMENTOS];
    bool es_vegetariano;
    int peso;
} lonchera_t;
```

Se pide

Crear una función que recorra una matriz no cuadrada de loncheras que representa al autobús y devuelva el índice de la columna en la que hay más loncheras que:

- Sean **vegetarianas**
- **y** que se llevaron para comer **ensalada**.
- **y** no hayan llevado **papas fritas**.

Ejercicio 2

El director de la orquesta de la escuela primaria de Springfield está descartando partituras para elegir cuál tocar en el próximo evento escolar.

Dado un vector de `partitura_t`, crear una función que lo recorra de **forma recursiva** y que devuelva la cantidad de partituras que puede descartar.

Una partitura se puede descartar si:

- tiene alguna nota cuyo sonido no sea válido.
- **o** su autor sea Chaikovski (es muy difícil de tocar).

```
typedef struct nota {
    char sonido[MAX_SONIDO];
    char figura_musical; // (R)edonda, (C)orchea, (S)emicorchea
    bool es_silencio;
} nota_t;

typedef struct partitura {
    nota_t notas[MAX_NOTAS];
    int cantidad_notas;
    char autor[MAX_AUTOR];
} partitura_t;
```

Aclaraciones

- Los sonidos válidos son "LA", "SI", "DO", "RE", "MI", "FA" y "SOL".