



Instituto Politécnico Nacional  
Unidad Profesional Interdisciplinaria en  
Ingeniería y Tecnologías Avanzadas



Programación Avanzada

Segundo Examen Parcial

Realice los programas que se solicitan. Una vez terminados, agregue sus archivos de código fuente a un archivo zip con nombre ApellidoPaterno ApellidoMaterno Nombre(s) E2.zip y súbalo a Classroom.

1. Escriba un programa en C que permita eliminar un elemento de un arreglo utilizando punteros, memoria dinámica y aritmética de punteros. El arreglo deberá permitir almacenar hasta  $n$  elementos generados de manera pseudo-aleatoria entre  $-(n \times 10)$  y  $(n \times 10)$ . Al inicio de su programa, deberá solicitar al usuario el valor de  $n$ . Enseguida, deberá mostrar el arreglo generado y solicitar al usuario el número que desee eliminar. En caso que existan números repetidos, se deberá eliminar el primero que se encuentre. En caso que el usuario proporcione un número que no existe en el arreglo, deberá mostrar un mensaje de error. El programa terminará cuando no queden elementos en el arreglo.

Ejemplo:

Número de elementos: 8

Arreglo generado: [1, 4, 6, 2, 7, 9, 5, 6]

Número a eliminar: 6

Arreglo resultante: [1, 4, 2, 7, 9, 5, 6]

Número a eliminar: 1

Arreglo resultante: [4, 2, 7, 9, 5, 6]

Número a eliminar: -9

Error: No se encontro el número.

Número a eliminar: 9

Arreglo resultante: [4, 2, 7, 5, 6]

Su programa deberá tener al menos las siguientes funciones:

```
void generar(int **arr, int *size);
```

Permite generar un arreglo de tamaño  $size$  e inicializarlo con números pseudo-aleatorios entre  $-(size \times 10)$  y  $(size \times 10)$ . Los parámetros se deberán pasar por referencia.

```
bool eliminar(int **arr, int *size, int num2del);
```

Permite eliminar un elemento de un arreglo. Deberá devolver como resultado true si la eliminación se ejecuta de forma exitosa, o false si no se encuentra el número a eliminar. El arreglo y el tamaño deberán pasarse por referencia. El número a eliminar deberá pasarse por valor.

```
void imprimir(int *arr, int size);
```

Permite imprimir un arreglo de tamaño  $size$ . Los parámetros deberán pasarse por

valor. Y dichas funciones deberán estar en una biblioteca de funciones.

