

# Academia de Tecnologías de Información y Telemática Centro de Nuevas Tecnologías

## LABORATORIO DE PROGRAMACIÓN APLICADA



	CARRERA	CICLO		ASIGNATURA		TEMA APLICADO
	ITI / ITEM	OTOÑO 2022	PR	PROGRAMACIÓN 2		Apuntadores
	No. PRÁCTICA	PARCIAL		VALOR EN PAR	CIAL	SESIONES DE DESARROLLO
1		Primer parcial		60%		1 sesión
1	OBJETIVO					

Aprender las bases de la programación en lenguaje C que permitan al alumno desarrollar su lógica en un entorno amigable, intuitivo y fácil de usar.

### 2 FUNDAMENTO TEÓRICO

#### MEMORIA DINÁMICA

Un puntero puede apuntar a un objeto de cualquier tipo y aunque no son lo mismo podemos manipularlo como un arreglo estático, para ello debemos reservar la memoria a la cual vamos a acceder, es decir, cuantos datos vamos a almacenar.

FUNCION MALLOC → puntero = (cast al tipo puntero) malloc(espacios \* sizeof (tipo de dato a almacenar);

Es recomendable verificar si la memoria se reservó correctamente, para ello comparamos si malloc nos regresó algo diferente a NULL, también se recomienda liberar memoria una vez que terminemos de utilizarla.

```
#include <stdio.h>
#include <stdib.h>

void main() {
   int *p = NULL;
   int nDatos = 10;

   p = (int *) malloc(nDatos * sizeof( int ));

   if (p != NULL)
        printf("\nSe ha asignado memoria correctamente\n\n");
        /*se puede meter aqui todo el desarrollo o para un mejor manejo de código hacerlo
        por separado pero comprobando si se asigno memoria*/
   else
        printf("\n\nNo se ha podido asignar menmoria\n");

        free(p); //Libera memoria para el puntero p
}
```

3	HERRAMIENTAS	4	RÚBRICA
	- Lenguaje DevC++		- Encabezado y documentación: requisito
	- Plataforma de evaluación		- Compilación: 10%
	- Conocimientos previos del tema		- Valor de evaluación indicado en cada punto: 90%

## 5 PLANTEAMIENTO Y DESARROLLO DE PRÁCTICA

#### Planteamiento:

Escribir un programa en C que genere aleatorio el tamaño de un vector (arreglo unidimensional) y que almacene caracteres aleatorios.

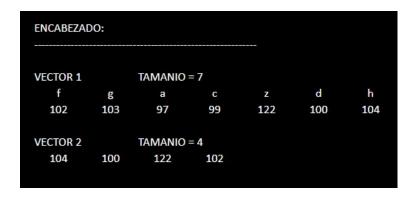
### **Requisitos:**

- 1. Para recorrer el vector se pide utilizar punteros en lugar de índices, el tamaño que se genera aleatoriamente esta entre 6 y 25.
- 2. Función llenadoA(): (4pts)
- Revise la sección de fundamento teórico para la reserva de memoria.
- Realiza la reserva de memoria y llena el vector con letras de la "a" a la "z" generadas de manera aleatoria.
- Imprime en pantalla el vector generado, indique el tamaño.
- Invoca a la función operación.

# 3. Función operación(): (4pts)

- Del vector obtiene aquellos caracteres que en su código ascii sean par, genera un vector dinámico entero para almacenar esos códigos de manera inversa.
- 4. Revise la sección 6 REQUISITOS DE ENTREGA

# SALIDA (2pts)



### 6 REQUISITOS DE ENTREGA

- ✓ Subir a blackboard únicamente el archivo .cpp
- ✓ Guardar el archivo .cpp con el siguiente nombre: MatriculaDelEstudiante\_P3P2A El archivo deberá estar en blackboard como máximo 10 minutos antes de concluir la sesión de laboratorio
- ✓ Sin excepción alguna, la liga no se reactivará una vez que se haya cerrado.
- ✓ Sin excepción alguna, el archivo de práctica deberá estar en la liga de blackboard para que pueda obtener una evaluación.
- ✓ No se aceptarán entregas fuera del horario establecido ni por otros medios.

7 RÚBRICA								
I.I. Luis Fernando Turrubiartes Email: <u>luis.turrubiartes@upslp.</u>		iviii. Eilialia Gallicz Zavala						
Profesor LAPLA	Profesor LAPLA	Micro academia de Programación	Coordinador Académico					