

CARRERA	CICLO	ASIGNATURA	TEMA APLICADO
ITI / ITEM	OTOÑO 2022	PROGRAMACIÓN 2	LISTAS SIMPLES
No. PRÁCTICA	PARCIAL	VALOR EN PARCIAL	SESIONES DE DESARROLLO
7	Segundo parcial	25%	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

LAS LISTAS

Son una estructura de datos, una secuencia conectada de nodos, cada uno de los cuales contiene algún dato.

- Hay un nodo al comienzo llamado la cabeza o frente (head o front, primero).
- Hay un nodo de término llamado la cola o atrás (tail o back, ultimo).
- Una Lista sólo puede ser recorrida en secuencia, usualmente hacia atrás o adelante.
- Recordando apuntadores, sabemos que estos pueden ser de cualquier tipo y nos ayudan a obtener los datos desde un espacio de memoria, por lo tanto, las listas son espacios en la memoria que pertenecen a una misma identidad y los unimos por medio del apuntador siguiente.

```
struct nodo{
    int dato;
    struct nodo *sig;
    struct nodo *ant
};
typedef struct nodo *Nod;
```

```
main( )
{
    Nod m_lista = NULL;
    int valor = 5;
    Inserta (m_lista, valor);
}
```

3 HERRAMIENTAS	4 RÚBRICA
<ul style="list-style-type: none"> - Lenguaje DevC++ - Plataforma de evaluación - Conocimientos previos del tema 	<ul style="list-style-type: none"> - Encabezado y documentación: requisito - Compilación: 10% - Valor de evaluación indicado en cada punto: 90%

5 PLANTEAMIENTO Y DESARROLLO DE PRÁCTICA

Planteamiento:

Analizar y comprender los apuntadores dentro de listas.

Requisitos:

1. Analice el siguiente código y escriba en una hoja la salida que mostrara.
2. La hoja debe contener también las operaciones o análisis que realizo para llegar al resultado.
3. En una hoja escriba su matrícula, nombre, grupo de laboratorio y fecha.
4. Indique cual es la salida en pantalla.

```

#define N 10
#define M 15
struct m_tipo{
    int value;
    char name[10];
    struct m_tipo* apunta_hacia;
}*primero = NULL, *ultimo = NULL, *select = NULL;

int iv = 0, in = 0, cv = 0, cn = 0;
char names[M][N] = {"Pato", "Perro", "Gato", "Cisne", "Raton", "Gaviota", "Zorro", "Borrego",
                    "Rana", "Paloma", "Tortuga", "Vaca", "Toro", "Gusano", "Cocodrilo"};
m_tipo* creaReg(){
    m_tipo *nuevo = new m_tipo; //equivalente a malloc reserva de memoria para el nuevo
    nuevo->value = cn;
    strcpy(nuevo->name, names[in]);
    nuevo->apunta_hacia = NULL;
    return nuevo;
}

void printList(){
    m_tipo *iterator = primero;

    while(iterator!=NULL){//impresión hacia la derecha
        printf("[%d, %s] --> ", iterator->value, iterator->name);
        iterator = iterator->apunta_hacia;
    }
}

void insert(){
    m_tipo *nuevo = new m_tipo, *pp = NULL, *del = NULL; //equivalente a malloc reserva de memoria para el nuevo
    nuevo->value = 5;
    strcpy(nuevo->name, "Pato");
    nuevo->apunta_hacia = NULL;
    primero = nuevo;
    ultimo = nuevo;
    select = nuevo;

    for(int i=0; i<N-2; i++){
        nuevo = creaReg(); //CREACION DEL NODO
        cn += 1, iv += 1, in += 1, cv += 1; //INCREMENTO DE VARIABLES
        if(nuevo->value < select->value){
            nuevo->apunta_hacia = select->apunta_hacia;
            select->apunta_hacia = nuevo;
        }else{
            pp = ultimo;
            ultimo->apunta_hacia = nuevo;
        }
        if(nuevo->apunta_hacia==NULL)
            ultimo = nuevo;
    }
    printList();
    for(int i=0; i<3; i++){
        if(pp->value > primero->value){
            del = primero->apunta_hacia;
            primero->apunta_hacia = del->apunta_hacia;
            printf("nodo del: %d,%s\n", del->value, del->name);
        }
    }
    printList();
}

main(){ insert();}

```

6 REQUISITOS DE ENTREGA

- ✓ Subir a blackboard únicamente el archivo .cpp
- ✓ Guardar el archivo .cpp con el siguiente nombre: **MatriculaDelEstudiante_P8P2A** 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 Turrubiarres Gómez Email: luis.turrubiarres@upslp.edu.mx	I.C. Atzel Yazmin Rivas Ortega Email: atzel.rivas@upslp.edu.mx	MTI. Liliana Gámez Zavala	Dr. Francisco Cruz Ordaz Salazar
Profesor LAPLA	Profesor LAPLA	Micro academia de Programación	Coordinador Académico