



**Universidad Nacional Autónoma  
de México**

**Facultad de Ingeniería**



**Semestre 2021-1**

**Estructura de Datos y Algoritmos**

**Actividad : Push/pop**

**Nombre del alumno: Perera Martínez**

**David**

**Fecha: 15 de Junio de 2021**

## ***Algoritmo correspondiente a la operación push y otro a la operación pop***

**1.-**

```
//main.c
#include <stdio.h>
#include <stdlib.h>
#include "pila.h"
int main()
{
    Pila *pilaLSE;
    int dato,opcion;
    pilaLSE = crearPila();

    printf("\t|PILA\n");
    do {
        printf("ingrese una opcion de las siguientes:\n");
        printf("1)push \n2)pop \n3)listar \n4)borrar pila \n5)Salir:\n");
        scanf("%d",&opcion);

        switch (opcion){
            case 1://push
                printf("\ninserte el dato a guardar: ");
                scanf("%d",&dato);
                push(pilaLSE,dato);
                system("cls");
                break;
            case 2://pop
                system("cls");
                dato=pop(pilaLSE);
                printf("dato eliminado: %d",dato);
                printf("\n---PILA---\n");
                listar(*pilaLSE);
                printf("\n");
                break;
            case 3://listar
                system("cls");
                printf("\n---PILA---\n");
                listar(*pilaLSE);
                printf("\n");

                break;
            case 4://liberar pila
                inicializarPila(pilaLSE);
```

```

        free(pilaLSE);
    break;
}
}while (opcion !=5);
return 0;
}

```

## 2.-

```

//pila.h
#ifndef PILA_H_INCLUDED
#define PILA_H_INCLUDED

typedef struct nodo {
    int info;
    struct nodo *sig; //APUNTADOR AL SIG NODO DE LA PILA
} Nodo;

typedef struct {
    Nodo *tope; //INDICA LA CIMA DE LA PILA
} Pila;

Pila *crearPila(void);

void push(Pila *pila,int dato);
void listar(Pila pila);
int pop(Pila *pila);
int pilaVacía(Pila pila);
void inicializarPila(Pila *pila);

#endif // PILA_H_INCLUDED

```

## 3.-

```

//pila.c
#include <stdio.h>
#include <stdlib.h>
#include "pila.h"

Pila *crearPila(){
    Pila *nuevaPila;
    //CREACIÓN DE LA PILA
    nuevaPila=malloc(sizeof(Pila));
}

```

```

if (nuevaPila == NULL){
    printf("\nError: Espacio insuficiente para crear la pila...");
    return NULL;
}

//INICIALIZACIÓN
nuevaPila->tope=NULL;
return nuevaPila;
}

void push(Pila *pila,int dato){
    //variables locales
    Nodo *nuevo;

    //1.-creación del nodo
    nuevo = calloc(1,sizeof(Nodo));
    if(nuevo==NULL){
        printf("\nERROR: Memoria del sistema insuficiente...");
        return;
    }
    //2.-inicializar nodo
    nuevo->info=dato;
    nuevo->sig=pila->tope;

    //3.-integrar a la pila
    pila->tope = nuevo;

}

void listar(Pila pila){
    //variables locales
    Nodo *q = pila.tope;
    while(q!=NULL){
        printf("%d\n",q->info);
        q=q->sig;
    }

}

int pop(Pila *pila){
    int aux;
    Nodo *q=pila->tope;
    pila->tope=pila->tope->sig;
    aux = q->info;
    free(q);
    return aux;
}

```

```
int pilaVacía(Pila pila){
    return pila.tope==NULL;
}
void inicializarPila(Pila *pila){
    Nodo *q = pila->tope;
    while(q!=NULL)
    {
        pila->tope=pila->tope->sig;
        free(q);
        q=pila->tope;
    }
    printf("\nPila vacía....");
}
```