



**UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO**



FACULTAD DE INGENIERIA

Estructuras de Datos y Algoritmos I

Actividad #2 “Push y Pop”

Alumno: García Gallegos Luis

Grupo:15

SEMESTRE 2021-2

Fecha de entrega 16/06/2021

Push en C

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #define p printf
4  #define s scanf
5
6  struct Nodo{
7      int dato;
8      struct Nodo *siguiente;
9  };
10 void agregarPila(struct Nodo *, int);
11 int main(){
12     struct Nodo *pila= NULL;
13     int n1,n2;
14
15     p("Digite un numero ");
16     s("%d",&n1);
17     agregarPila(pila,n1);
18
19     p("Digite otro numero ");
20     s("%d",&n2);
21     agregarPila(pila,n2);
22
23 }
24 void agregarPila(struct Nodo *pila, int n){
25     struct Nodo *nuevo_nodo = (struct Nodo*) malloc(sizeof(struct Nodo));
26     nuevo_nodo->dato = n;
27     nuevo_nodo->siguiente = pila;
28     pila = nuevo_nodo;
29     p("\nElemento %d agregado a la Pila correctamente \n", n);
30 }
```

```

2  #include <stdlib.h>
3  #define p printf
4  #define s scanf
5
6  struct Nodo{
7      int dato;
8      struct Nodo *siguiente;
9  };
10 void agregarPila(struct Nodo *, int);
11 int main(){
12     struct Nodo *pila= NULL;
13     int n1,n2;
14
15     p("Digite un numero ");
16     s("%d",&n1);
17     agregarPila(pila,n1);
18
19     p("Digite otro numero ");
20     s("%d",&n2);
21     agregarPila(pila,n2);
22
23 }
24 void agregarPila(struct Nodo *pila, int n){
25     struct Nodo *nuevo_nodo = (struct Nodo*) malloc(sizeof(struct Nodo));
26     nuevo_nodo -> dato = n;
27     nuevo_nodo->siguiente = pila;
28     pila = nuevo_nodo;
29     p("\nElemento %d agregado a la Pila correctamente \n", n);
30 }

```

Digite un numero 5

Elemento 5 agregado a la Pila correctamente

Digite otro numero 9

Elemento 9 agregado a la Pila correctamente

Process exited after 4.14 seconds with return value 46

Presione una tecla para continuar . . . █

Push en C++

```
1  #include <iostream>
2  #include <conio.h>
3  #include <stdlib.h>
4
5  using namespace std;
6
7  struct Nodo{
8      int dato;
9      Nodo *siguiente;
10 };
11 void agregarPila(Nodo *&, int);
12 int main(){
13     Nodo *pila= NULL;
14     int n1,n2;
15
16     cout<<"Digite un numero ";
17     cin>>n1;
18     agregarPila(pila,n1);
19
20     cout<<"Digite otro numero ";
21     cin>>n2;
22     agregarPila(pila,n2);
23
24     getch();
25     return 0;
26 }
27 void agregarPila(Nodo *&pila, int n){
28     Nodo *nuevo_nodo = new Nodo();
29     nuevo_nodo ->dato = n;
```

```

5   using namespace std;
6
7   struct Nodo{
8       int dato;
9       Nodo *siguiente;
10  };
11  void agregarPila(Nodo *&, int);
12  int main(){
13      Nodo *pila= NULL;
14      int n1,n2;
15
16      cout<<"Digite un numero ";
17      cin>>n1;
18      agregarPila(pila,n1);
19
20      cout<<"Digite otro numero ";
21      cin>>n2;
22      agregarPila(pila,n2);
23
24      getch();
25      return 0;
26  }
27  void agregarPila(Nodo *&pila, int n){
28      Nodo *nuevo_nodo = new Nodo();
29      nuevo_nodo->dato = n;
30      nuevo_nodo->siguiente = pila;
31      pila = nuevo_nodo;
32      cout<<"\nElemento " <<n<<" agregado a Pila correctamente |"<<endl;
33  }

```

Digite un numero 3

Elemento 3 agregado a Pila correctamente

Digite otro numero 7

Elemento 7 agregado a Pila correctamente

Process exited after 6.046 seconds with return value 0

Presione una tecla para continuar . . .

Pop en C

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #define p printf
4  #define s scanf
5
6  struct Nodo{
7      int dato;
8      struct Nodo *siguiente;
9  };
10 void agregarPila(struct Nodo *, int);
11 void sacarPila(struct Nodo *, int);
12 int main(){
13     struct Nodo *pila = NULL;
14     int dato;
15
16     p("Digite un numero: ");
17     s("%d",&dato);
18     agregarPila(pila,dato);
19
20     p("Digite otro numero: ");
21     s("%d",&dato);
22     agregarPila(pila,dato);
23
24     p("\nSacando los elementos de la Pila ");
25     while(pila != NULL){
26         sacarPila(pila,dato);
27         if (pila != NULL){
28             p("%d , ", dato);
29         }
```

```

21     s("%d",&dato);
22     agregarPila(pila,dato);
23
24     p("\nSacando los elementos de la Pila ");
25     while(pila != NULL){
26         sacarPila(pila,dato);
27         if (pila != NULL){
28             p("%d , ", dato);
29         }
30         else{
31             p("%d.", dato);
32         }
33     }
34     getch();
35     return 0;
36 }
37 void agregarPila(struct Nodo *pila, int n){
38     struct Nodo *nuevo_nodo = (struct Nodo*) malloc(sizeof(struct Nodo));
39     nuevo_nodo->dato = n;
40     nuevo_nodo->siguiente = pila;
41     pila = nuevo_nodo;
42     p("\tElemento %d ha sido agregado a PIL correctamente \n", n);
43 }
44 void sacarPila(struct Nodo *pila, int n){
45     struct Nodo *aux = pila;
46     n = aux->dato;
47     pila = aux->siguiente;
48     free(aux);
49 }

```

Pop en C++


```
1  #include<iostream>
2  #include<conio.h>
3
4  using namespace std;
5
6  struct Nodo{
7      int dato;
8      Nodo *siguiente;
9  };
10
11 void agregarPila(Nodo *&,int); //Prototipo de función
12 void sacarPila(Nodo *&,int &);
13
14 int main (){
15     Nodo *pila = NULL;
16     int dato;
17     cout <<"Digite un numero: ";
18     cin>>dato;
19     agregarPila(pila,dato);
20
21     cout <<"Digite otro numero: ";
22     cin>>dato;
23     agregarPila(pila,dato);
24
25     cout<<"\nSacando los elementos de la Pila |";
26
27     while(pila != NULL){
28         sacarPila(pila,dato);
29         if(pila != NULL){
```



```

26
27 while(pila != NULL){
28     sacarPila(pila,dato);
29     if(pila != NULL){
30         cout<< dato <<" , ";
31     }
32     else{
33         cout<< dato <<".";
34     }
35 }
36
37 getch();
38 return 0;
39 }
40 void agregarPila(Nodo *&pila,int n){
41     Nodo *nuevo_nodo = new Nodo(); //1. Crear el espacio en memoria para almacenar un nodo.
42     nuevo_nodo->dato = n; //2. Cargar el valor dentro del nodo (dato).
43     nuevo_nodo->siguiente = pila; //3. Cargar el puntero pila dentro del nodo (*siguiente)
44     pila = nuevo_nodo; //4. Asignar el nuevo nodo a la pila.
45
46     cout<<"\tElemento " <<n<<" ha sido agregado a PIL correctamente" <<endl;
47 }
48
49 void sacarPila(Nodo *&pila, int &n) {
50     Nodo *aux = pila;
51     n = aux->dato;
52     pila = aux->siguiente;
53     delete aux;
54 }

```

Digite un numero: 12

Elemento 12 ha sido agregado a PIL correctamente

Digite otro numero: 15

Elemento 15 ha sido agregado a PIL correctamente

Sacando los elementos de la Pila 15 , 12.

Process exited after 10.88 seconds with return value 0

Presione una tecla para continuar . . .