

1. Realizar un programa en c++ para implementar los siguientes módulos:

Menú de Opciones
1. PALINDROMO
2. BORRAR DATO DE PILA
3. SALIR

Utilizar pilas y colas en la solución **(10 Pts.)**

SOLUCION:

```
#include <iostream>
#include<stdlib.h>
#include<string.h>
using namespace std;

/*estructura de un nodo*/
struct nodo{
int dato;
nodo *siguiente;
};

//PROTOTIPOS DE FUNCIONES Y PROCEDIMIENTOS
void menu();
void ponerpila(nodo *&pila,char n);
void sacarpila(nodo *&pila,char &n);
void ponerpila(nodo *&pila,int n);
void sacarpila(nodo *&pila,int &n);
bool cola_vacia(nodo *frente);
void insertarcola(nodo *&frente, nodo *&fin,char n);
void sacarcola(nodo *&frente,nodo *&fin,char &n);
void palindromo(nodo *&pila,nodo *&frente,nodo *&fin);
void borrar dato(nodo *&pila,int n);

//PROGRAMA PRINCIPAL
int main()
{
    int opcion;
    char cade[15];
    int tam,i,dato,n1;
    char c,op;
    do{
        nodo *pila=NULL;
        nodo *frente=NULL;
        nodo *fin=NULL;
        i=0;
        c=' ';
        menu();
        cin>> opcion;
        switch(opcion){
        case 1: cout<<"\nPALINDROMO\n ";
                cout<<"\nAGREGANDO DATOS EN LA PILA Y LA COLA\n ";
                cout << "\nIngresa una palabra ";
                cin>>cade;
                tam=strlen(cade);
                while(i<tam ){
                    c=cade[i];
                    ponerpila(pila,c);
                    insertarcola(frente,fin,c);
                    i++;
                }
                palindromo(pila,frente,fin);
                break;
```

```
case 2 :do{
    cout<<"\nAPILANDO DATOS EN LA PILA\n ";
    cout << "\nIngrese dato para poner en la pila ";
    cin>>dato;
    ponerpila(pila,dato);
    cout<<"MAS DATOS?-->s/n ";
    cin>>op;
}while(op!='n');
cout<<"QUE DATO QUIERE BORRAR?"<<endl;
cin>>n1;
borradato(pila,n1);
if(pila!=NULL){
    cout<<"\nLA PILA CON DATO N BORRADO ES \n ";
    cout<<" PILA"<<endl;
    while(pila!=NULL){
        sacarpila(pila,dato);
        cout<<" "<<dato<<endl;
        cout<<" ----"<<endl;
    }
}
else{
    cout<<"PILA ESTA VACIA!!!";
}
break;
case 3 :break;
}
}while(opcion!=3);
return 0;
}
```

```
/*PROCEDIMIENTO MENU*/
void menu(){
    cout<<"\n-----\n";
    cout<<"    MENU DE OPCIONES    \n";
    cout<<" -----\n";
    cout<<"    1- COPIA        \n";
    cout<<"    2- BORRAR PILA    \n";
    cout<<"    3- Salir         \n";
    cout<<"-----\n";
    cout<<" Que desea hacer? ";
}
}
```

```
/*procedimiento para poner (push)datos a la pila palindromo*/
void ponerpila (nodo *&pila,char n){
    nodo *nuevo_nodo= new nodo();//paso1
    nuevo_nodo->dato=n; //paso 2
    nuevo_nodo->siguiente=pila;
    pila=nuevo_nodo;//paso 3
}
}
```

```
/*procedimiento para sacar (pop)datos a la pila palindromo*/
void sacarpila(nodo *&pila,char &n){
    nodo *aux=pila;
    n=aux->dato;
    pila=aux->siguiente;
    delete aux;
}
}
```

```
/*procedimiento para poner (push)datos a la pila*/
void ponerpila (nodo *&pila,int n){
    nodo *nuevo_nodo= new nodo();//paso1
    nuevo_nodo->dato=n; //paso 2
    nuevo_nodo->siguiente=pila;
    pila=nuevo_nodo;//paso 3
}
```

```
/*procedimiento para sacar (pop)datos a la pila*/
void sacarpila(nodo *&pila,int &n){
    nodo *aux=pila;
    n=aux->dato;
    pila=aux->siguiente;
    delete aux;
}
```

```
/*FUNCION PARA VER SI LA COLA ESTA VACIA*/
bool cola_vacia(nodo *frente){
    if(frente==NULL)
        return true;
    else
        return false;
}
```

```
/*PROCEDIMIENTOS PARA INSERTAR UN DATO EN LA COLA*/
void insertarcola (nodo *&frente, nodo *&fin,char n){
    nodo *nuevo_nodo= new nodo();
    nuevo_nodo->dato=n;
    nuevo_nodo->siguiente=NULL;
    if(cola_vacia(frente)){
        frente=nuevo_nodo;
    }
    else{
        fin->siguiente=nuevo_nodo;
    }
    fin=nuevo_nodo;
}
```

```
/*PROCEDIMIENTO PARA QUITAR UN DATO A LA COLA*/
void sacarcola(nodo *&frente,nodo *&fin,char &n){
    nodo *aux=frente;
    n=frente->dato;
    if(frente==fin)
    {
        frente=NULL;
        fin=NULL;
    }
    else
    {
        frente=frente->siguiente;
    }
    delete aux;
}
```

```
/*PROCEDIMIENTO PARA VERIFICAR PALINDROMO*/
void palindromo(nodo *&pila,nodo *&frente,nodo *&fin){
char dato=' ',dato1=' ';
if(pila!=NULL && frente!= NULL){
    while(pila!=NULL && frente!= NULL && dato==dato1){
        sacarpila(pila,dato);
        sacarcola(frente,fin,dato1);
        cout<<" "<<dato<<" | "<<dato1<<endl;
        cout<<" ----"<<" | "<<" ---"<<endl;
    }
}
else{
    cout<<"PILA O COLA ESTA VACIA!!!";
}
if(dato1==dato)
cout<<"PALABRA ES PALINDROMO!!!"<<endl;
else
cout<<"PALABRA NO ES PALINDROMO!!!"<<endl;
}
/*procedimiento para borrar dato de pila*/
void borrarato(nodo *&pila,int n){
nodo *auxp=NULL;
int i=1,d=0;
while(pila!=NULL && i<n){
    i++;
    sacarpila(pila,d);
    ponerpila(auxp,d);
}
sacarpila(pila,d);
while(auxp!=NULL){
    sacarpila(auxp,d);
    ponerpila(pila,d);
}
}
```