



**Instituto
Politécnico
Nacional**



REPORTE DE PRACTICA

P21

SIMULAR LAS VIAS DEL TREN

Julio Cesar Hernández Reyes

Grupo:

2CV11

Profesor:

Juan Jesús Gutiérrez García

Algoritmos y Estructura de Datos

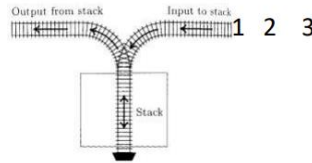
Noviembre-2020

OBJETIVO:

Construir una función main para ejemplificar el uso de una pila de enteros simulando el problema de las vías del tren.

Problema:

Un conjunto de trenes está del lado derecho de la vía, figura siguiente, digamos 1,2,3 y deben pasar al lado izquierdo en un orden específico.



Por ejemplo, si el orden final deseado es 1 3 2 entonces habrá que hacer las siguientes operaciones

1. Pasar el tren 1 a la pila (push)
2. Sacar el tren 1 de la pila (pop) /* salida 1*/
3. Pasar el tren 2 a la pila (push)
4. Pasar el tren 3 a la pila (push)
5. Sacar el tren 3 de la pila (pop) /* salida 3*/
6. Sacar el tren 2 de la pila (pop) /* salida 2*/

Lo que debes realizar en esta práctica es encontrar el orden en que se deben hacer las operaciones en la pila, la función pop se acompaña de mostrar el tope en la salida.

ELEMENTOS DE LA PRACTICA

En esta práctica sólo debes modificar la función main ajustándote a las restricciones que se listan a continuación. En el archivo P21.ZIP encontraras

- a) Archivo usa_pila.c, que contiene función main
- b) Archivos cabecera: pila.h y err.h con la definición de los prototipos, macros y constantes.
- c) Archivo con código objeto: pila.o (Versión de Windows)

RESTRICCIONES PARA MODIFICAR LA FUNCIÓN MAIN.

- a) El código que agregues sólo debe estar entre los comentarios que señalan dónde inicia tu código y dónde termina
- b) No debes declarar más variables
- c) Para compilar el programa usa el comando:
`gcc usa_pila.c pila.o`
- d) Cada vez que invoques a la función `tope_pil` debes también mostrar el valor del tope en pantalla, usando la función `printf`
- e) Puedes usar la función `tostr_pil` únicamente para depurar tu código en la versión final no debe aparecer en ningún momento.
- f) Al ejecutarse el programa sólo mostrará la cadena destino, así

G:\>a

1 3 2

RESULTADOS DE LA SEMANA DE PRACTICA CON EL CODIGO MAIN USANDO DIFERENTES CANTIDAD DE NUMEROS Y TAMBIEN EL ORDEN REQUERIDO(Resumen a tres numeros)

Cantidad de Números: 3

Orden Inicial: 1,2,3

Salida requerida:	Se puede realizar este ordenamiento. SI o NO. ¿Por qué?	Como se realiza el ordenamiento
123	SI	<pre> for(i=0;i<TAM_ENT;i++){ /*Inicia aqui tu codigo*/ switch(i){ case 0: push_pil(&p,entrada[i]); tope_pil(&tope,&p); printf(" %d ",tope); pop_pil(&p); break; case 1: push_pil(&p,entrada[i]); tope_pil(&tope,&p); printf(" %d ",tope); pop_pil(&p); break; case 2: push_pil(&p,entrada[i]); tope_pil(&tope,&p); printf(" %d ",tope); pop_pil(&p); break; } /*Termina aqui tu codigo*/ } </pre>
321	SI	<pre> for(i=0;i<TAM_ENT;i++){ /*Inicia aqui tu codigo*/ switch(i){ case 0: push_pil(&p,entrada[i]); break; case 1: push_pil(&p,entrada[i]); break; case 2: push_pil(&p,entrada[i]); tope_pil(&tope,&p); printf(" %d ",tope); pop_pil(&p); tope_pil(&tope,&p); printf(" %d ",tope); pop_pil(&p); tope_pil(&tope,&p); printf(" %d ",tope); pop_pil(&p); break; } /*Termina aqui tu codigo*/ } </pre>
312	NO Al momento de empezar a acomodar los trenes para que el primero en salir sea el 3 todo va bien porque el 3 es el que está en el tope, pero al momento de querer sacar el 1 no se puede porque el que está en el tope es el 2 no el 1. Entonces no se puede cumplir este ordenamiento.	
213	SI	<pre> for(i=0;i<TAM_ENT;i++){ /*Inicia aqui tu codigo*/ switch(i){ case 0: push_pil(&p,entrada[i]); break; case 1: push_pil(&p,entrada[i]); tope_pil(&tope,&p); printf(" %d ",tope); pop_pil(&p); tope_pil(&tope,&p); printf(" %d ",tope); pop_pil(&p); break; case 2: push_pil(&p,entrada[i]); tope_pil(&tope,&p); printf(" %d ",tope); pop_pil(&p); break; } /*Termina aqui tu codigo*/ } </pre>

RESULTADOS CON LAS SERIES SOLICITADAS EN LA LISTA

No	Nombre	Programa 1		Programa 2		Programa 3	
		Origen	Destino	Origen	Destino	Origen	Destino
68	Julio Cesar Hernández Reyes	1 2 3 4 5 6 7	3 2 5 6 4 1 7	1 2 3 4 5 6 7	3 4 2 1 5 7 6	1 2 3 4 5 6 7	3 5 4 6 7 2 1

Simbolo del sistema

```
C:\Users\Cesar Hernández\Desktop\AlgoEstrucDat\SegundoParcial\P21\P21>gcc serie1.c pila.o
C:\Users\Cesar Hernández\Desktop\AlgoEstrucDat\SegundoParcial\P21\P21>a
Cadena Inicial:
1 2 3 4 5 6 7
Cadena Final:
3 2 5 6 4 1 7

C:\Users\Cesar Hernández\Desktop\AlgoEstrucDat\SegundoParcial\P21\P21>gcc serie2.c pila.o
C:\Users\Cesar Hernández\Desktop\AlgoEstrucDat\SegundoParcial\P21\P21>a
Cadena Inicial:
1 2 3 4 5 6 7
Cadena Final:
3 4 2 1 5 7 6

C:\Users\Cesar Hernández\Desktop\AlgoEstrucDat\SegundoParcial\P21\P21>gcc serie3.c pila.o
C:\Users\Cesar Hernández\Desktop\AlgoEstrucDat\SegundoParcial\P21\P21>a
Cadena Inicial:
1 2 3 4 5 6 7
Cadena Final:
3 5 4 6 7 2 1
```

```
for(i=0;i<TAM_ENT;i++){
    /*Inicia aqui tu codigo*/
    switch(i){
        case 0:
            push_pil(&p,entrada[i]);
            break;
        case 1:
            push_pil(&p,entrada[i]);
            break;
        case 2:
            push_pil(&p,entrada[i]);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            break;
        case 3:
            push_pil(&p,entrada[i]);
            break;
        case 4:
            push_pil(&p,entrada[i]);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            break;
        case 5:
            push_pil(&p,entrada[i]);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            break;
        case 6:
            push_pil(&p,entrada[i]);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            break;
    }
    /*Termina aqui tu codigo*/
}
```

```
for(i=0;i<TAM_ENT;i++){
    /*Inicia aqui tu codigo*/
    switch(i){
        case 0:
            push_pil(&p,entrada[i]);
            break;
        case 1:
            push_pil(&p,entrada[i]);
            break;
        case 2:
            push_pil(&p,entrada[i]);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            break;
        case 3:
            push_pil(&p,entrada[i]);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            break;
        case 4:
            push_pil(&p,entrada[i]);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            break;
        case 5:
            push_pil(&p,entrada[i]);
            break;
        case 6:
            push_pil(&p,entrada[i]);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            break;
    }
    /*Termina aqui tu codigo*/
}
```

```
for(i=0;i<TAM_ENT;i++){
    /*Inicia aqui tu codigo*/
    switch(i){
        case 0:
            push_pil(&p,entrada[i]);
            break;
        case 1:
            push_pil(&p,entrada[i]);
            break;
        case 2:
            push_pil(&p,entrada[i]);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            break;
        case 3:
            push_pil(&p,entrada[i]);
            break;
        case 4:
            push_pil(&p,entrada[i]);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            break;
        case 5:
            push_pil(&p,entrada[i]);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            break;
        case 6:
            push_pil(&p,entrada[i]);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            tope_pil(&tope,&p);
            printf(" %d ",tope);
            pop_pil(&p);
            break;
    }
    /*Termina aqui tu codigo*/
}
```

Conclusion:

Las tres series que me tocaron hacer a mi se pudieron realizar de forma exitosa, para esto yo estuve usando un switch adentro del for por lo que no rompi ninguna regla de las que nos especifico entonces creo que pude lograr facilmente el ordenamiento de esos 7 numeros. En si me gusto esta practica porque fue entretenida y aprendi lo basico de las pilas que creo las usaremos mas seguido de ahora en adelante.