

Trabajo Práctico N° 9 Estructuras Dinámicas: Pila y Cola.

Objetivo: Interiorizarnos en la creación y en el manejo de estructuras Dinámicas (Pila y Cola) definidas por desarrollador.

PILA

Una **PILA** es una estructura de datos dinámica cuyos elementos se manipulan siguiendo una política **LIFO**: last-in, first-out, es decir el último dato almacenado es el primero en ser sacado y procesado. De esta forma, los elementos de una pila son almacenados y eliminados de la misma por un extremo común o 'tope'.

Ejemplo: Se almacena una palabra en una pila, de a una letra, y se desea imprimir la palabra invertida.

Programa En Pseudocódigo:

Programa PILA

```
// Programador: Gabriela Cerra  
// Fecha: 10 de Octubre de 2009  
// Versión: 01
```

Tipos estructurados

```
Pila = ^ elem  
Elem = registro  
Dato: carácter 1  
Psig: pila  
Finregistro
```

Procedimiento crear (ref P: pila)

Hacer

```
P := nil
```

Finhacer

Finprocedimiento

Procedimiento Apilar (ref P: pila; ele: carácter 1)

Variables

```
Nuevo: pila
```

Hacer

```
Crear (nuevo)  
Nuevo^. Dato := ele  
Nuevo^.psig := P  
P := nuevo
```

Finhacer

Finprocedimiento

Procedimiento Desapilar (ref P: pila; ref ele: carácter 1)

Variables

```
aux: pila
```

Hacer

```
Aux := p /* guardo la dirección de inicio de la pila */  
Ele := aux^.dato /* retorno el elemento del tope de la pila */  
P := p^.psig /* se corre el inicio de la pila */
```

Liberar (aux) /* se libera el nodo que estaba en el tope*/

Finhacer

Finprocedimiento

Función Esvacia (p: pila): booleana

Hacer

Si p= nil entonces esvacia:= verdadero

sino esvacia:= falso

finsi

Finhacer

Finfuncion

Variables

P: pila

Let: carácter 1

Hacer

CrearPila (p)

Imprimir: 'ingrese una letra de la palabra'

Leer: let

Repetir mientras (let <> ' ')

Apilar (p, let)

Imprimir: 'ingrese una letra de la palabra'

Leer: let

Finrepetirmientras

Imprimir: 'palabra invertida'

Repetir mientras (not (espacia(p)))

Desapilar (p, let)

Imprimir: let

Finrepetirmientras

Finhacer

Finprograma.

Ejercicios de Pila

1. Lea una palabra e imprima un mensaje indicando si es palíndromo o no. Use pilas. Una palabra es palíndromo cuando se lee igual hacia adelante que hacia atrás. Ejemplo: oso, radar, reconocer, rotor, seres, somos, etc.
2. Un conductor maneja de un pueblo origen a un pueblo destino, pasando por varios pueblos. Una vez en el pueblo destino, el conductor debe regresar a casa por el mismo camino. Muestre el camino recorrido tanto de ida como de vuelta.
3. Lea una cadena y determine si los símbolos () y [] están correctamente balanceados. Si no lo están muestre el error indicando el símbolo faltante. Use una pila para cada símbolo.
4. Un almacén tiene capacidad para apilar n contenedores. Cada contenedor tiene un número de identificación. Cuando se desea retirar un contenedor específico, deben retirarse primero los contenedores que están encima de él y colocarlos en otra pila, efectuar el retiro y regresarlos. Codifique los métodos Push () y Pop () para gestionar los contenedores.

COLA

Una **COLA** es una estructura de datos dinámica cuyos elementos se manipulan siguiendo una política **FIFO**: first-in, first-out, es decir el primer dato almacenado es el primero en ser sacado y procesado. De esta forma, los elementos de una cola son almacenados y eliminados de la misma por extremos opuestos.

Ejemplo: En una sala de espera de un consultorio se encuentran los pacientes que van a visitar a un médico. Simular la atención de los mismos respetando el orden de llegada y determinar cuántos de ellos tiene obra social.

Programa En Pseudocódigo:

Programa COLA

```
// Programador: Gabriela Cerra  
// Fecha: 10 de Octubre de 2009  
// Versión: 01
```

Tipos estructurados

```
lista= ^ nodo  
    Nodo= registro  
    dato: carácter 20  
    obra: carácter 2  
    Psig: pila  
    Finregistro  
Cola= registro  
    Pin: lista  
    Pfin: lista  
    Finregistro
```

Procedimiento crearCola (ref C: cola)

```
Hacer  
c. pin:= nil  
c.pfin:= nil  
Finhacer  
Finprocedimiento
```

Procedimiento encolar (ref C: cola; ele: carácter 20; os: carácter 2)

Variables

Nuevo: lista

```
Hacer  
Crear (nuevo)  
Nuevo^. dato:= ele  
Nuevo^. Obra:= os  
Nuevo^.psig:= nil
```

```
Si c.pin = nil entonces c.pin= nuevo  
Sino c.pfin^. psig:= nuevo  
Finsi
```

```
c.pfin:= nuevo  
Finhacer  
Finprocedimiento
```

Procedimiento Desencolar (ref c: cola; ref ele: carácter 20; ref os: carácter 2)

Variables

aux: lista

Hacer

Aux:= c.pin

Ele:= aux^.dato

Os:= aux^.obra

c.pin:= aux^.psig

Si c.pin= nil entonces

c.pfin:= nil

finsi

Liberar (aux)

Finhacer

Finprocedimiento

Función Esvacia (c: cola): booleana

Hacer

Si c.pin= nil entonces esvacia:= verdadero

sino esvacia:= falso

Finsi

Finhacer

Finfuncion

Variables

Co: cola

Nom: carácter 20

Obsoc: carácter 2

C: entero 2

Hacer

CrearCola (co)

Imprimir: 'ingrese nombre de paciente 0 zzz para terminar'

Leer: nom

Repetir mientras (nom <> 'zzz')

Imprimir: 'ingrese si el paciente tiene obra social (si/no)'

Leer: obsoc

Encolar (co, nom, obsoc)

Imprimir: ' ingrese nombre de paciente 0 zzz para terminar'

Leer: nom

Finrepetirmientras

C:=0

Repetir mientras (not (esvacia (co)))

Desencolar (co, nom, obsoc)

Si obsoc = 'si' entonces c:= c+ 1

Finsi

Finrepetirmientras

Imprimir: 'hay', c, 'pacientes con obra social'

Finhacer

Finprograma.

Ejercicios de Cola

1. Se tiene una cola donde los hombres son muy traviesos, es por esto que se ordena dividir la cola en una de hombres y una de mujeres. Escriba un programa que haga esto.
2. Se tiene una cola en la cual cada elemento tiene una prioridad entre tres posibles: 1, 2 ó 3, siendo 1 la prioridad más alta y es la que se debe atender primero. Implemente este tipo de cola denominada Cola con Prioridad. Note que cuando un cliente llega a la cola debe ubicarse de tal modo que todos los que estén delante de él tengan mayor o igual prioridad.
3. Se tiene una cola en la cual se han repartido tickets con el orden de atención. Sin embargo, llegada la hora de inicio hay muchos “colados”, es por esto que se le ordena al vigilante que retire a todos aquellos que no tienen ticket. Muestre la cola inicial, qué elementos fueron retirados de la cola y la cola final. Sugerencia: desencole cada elemento, si tiene ticket se vuelve a encolar, sino se retira. Note que los elementos se retiran y se reingresan según alguna condición. A este tipo de cola se le llama Cola con Reingreso.
4. Una Cola Doble es aquella en la que se pueden insertar y eliminar elementos por cualquiera de sus extremos. Escriba un menú para gestionar esta cola.