Listas circulares

-Es una lista en la que cada nodo tiene un sucesor y un predecesor

-el sucesor del “ultimo” es el “primero”

-el predecesor del “primero” es el “ultimo”

-Iniciando en cualquier nodo, puede recorrer la lista completa

-el “first” apunta al “ultimo elemento”.

-Son útiles cuando se necesita acceder rápidamente el inicio/fin de la lista.

-El recorrido se detiene cuando el current es igual al first

Pilas

-Conjunto de elementos colocados uno sobre el otro

-Únicamente se puede acceder el ultimo elemento insertado

-Siempre se inserte al final (el final se llama tope)

-Naturaleza LIFO(Last input,first out put)

Tiene 3 operaciones:

-Push: insertar un elemento en el tope.

-Pop : Eliminar el elemento en el tope

-Peek: Ver el elemento en el tope

-se puede implementar con listas o con Arrays

Class Stack{

Private int maxsize;

Private Object[] stackarray;

Private int TopPosition = -1;

Void Push(Object new Object){

If(topPosition<maxSize){

Stackarray[++topPosition] = newObject;

}

}

Object pop(){

Return stackarray(topPosition--)

}

}

COLAS

Queue

Queue

Dequeue