What is a stack

What is Stack Data Structure ?

Definition : *Stack is a linear data structure which operates in a LIFO(Last In First Out) or FILO (First In Last Out) pattern.*

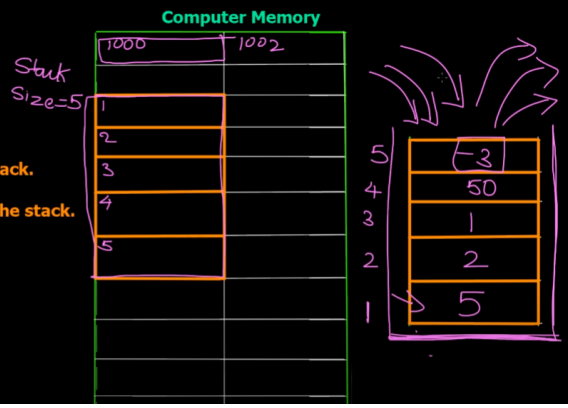
- It is named stack as it behaves like a real-world stack, for example – a deck of cards or a pile of plates, etc.
- Stack is an abstract data type with a bounded (predefined) capacity.
- It is a simple data structure that allows adding and removing elements in a particular order.
- The order may be LIFO(Last In First Out) or FILO(First In Last Out).

Working of Stack -

Stack Data Structure operates in a LIFO(Last In First Out) pattern or FILO (First In Last Out) pattern.

>> Items are added on top of the stack. This is know as PUSH operation

>> Items are removed from top of the stack. This is know as POP operation



Standard Stack Operations -

int Stack = Size(5)

1) push() -
Place an item onto the stack. If there is no place for new item, stack is in overflow state.

2) pop() -
Return the item at the top of the stack and then remove it.
If pop is called when stack is empty, it is in an underflow state.

3) isEmpty() -
Tells if the stack is empty or not

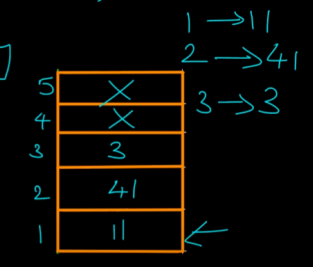
4) isfull() -
Tells if the stack is full or not.

5) peek() -
Access the item at the i position

6) count() -
Get the number of items in the stack.

7) change() -
Change the item at the i position

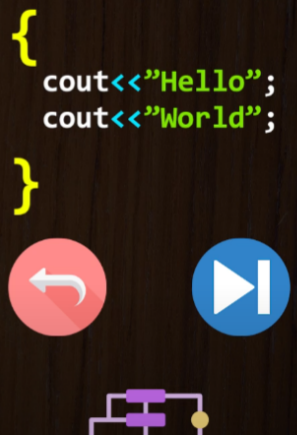
8) display() -
Display all items in the stack



change(1) = 11

Some Applications of Stack Data Structure

- Balancing of symbols
- Infix to Postfix /Prefix conversion
- Redo-undo features at many places like editors, photoshop.
- Forward and backward feature in web browsers
- Used in many algorithms like Tower of Hanoi, tree traversals, stock span problem, histogram problem.
- Other applications can be Backtracking, Knight tour problem, rat in a maze, N queen problem and sudoku solver



What is a Queue

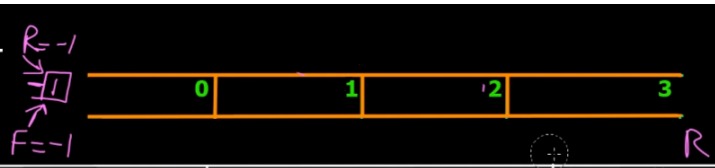
What is Queue Data Structure ?

Definition : *Queue is a linear data structure which operates in a First IN First OUT or Last IN Last OUT.*

- It is named queue as it behaves like a real-world queue, for example – queue(line) of cars in a single lane, queue of people waiting at food counter etc.
- Queue is an abstract data type with a bounded (predefined) capacity.
- It is a simple data structure that allows adding and removing elements in a particular order.
- The order is *FIFO(First IN First OUT)* or *LILO(Last In Last Out).*

3:29 / 26:37

Initialization -
int arr[4];
int rear = -1
int front = -1




```
isEmpty()
{
   if(front==-1 && rear == -1)
      return true
   else
      return false
}

isFull()
{
   if(rear == size(arr)-1)
      return true;
   else
      return false;
}
```

```
enqueue(value)
{
  if(isFull())
  return;
  else if(isEmpty)
  {
     rear = front = 0
  }
  else
  {
    rear++
  }
   arr[rear] = value
 }
```

```
dequeue()
{
  int x = 0;
  if(isEmpty())
  return;
  else if(front == rear)
  {
    x = arr[front]
    front = rear = -1
  }
  else
  {
    x = arr[front]
    front ++
  }
  return x;
}
```

Some Applications of Queue Data Structure

Queue is used when things but have to be processed in First In First Out order. Like –

- CPU scheduling, Disk Scheduling.
- Handling of interrupts in real-time systems. The interrupts are handled in the same order as they arrive, First come first served.
- In real life, Call Center phone systems will use Queues, to hold people calling them in an order, until a service representative is free.
- When data is transferred asynchronously between two processes. Queue is used for synchronization.