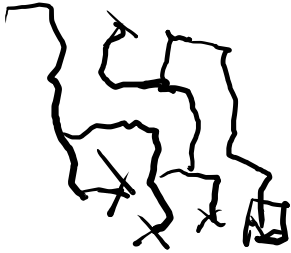




Stack



DFS
depth first

Array Stack

```

graph LR
    stack[stack] --> top[top]
    top --> array[ ]
  
```

List Stack

```

graph LR
    top((top)) --> next[next]
    next --> node1[ ]
    node1 --> node2[ ]
    node2 --> node3[ ]
    node3 --> node4[ ]
  
```

list NODE

```

graph LR
    node[ ] -- next --> next[ ]
    node -- value --> val((25))
  
```

push (int e)

- make new node w/ e inside the node
- make it point to top
- place it at top

pop (int e)

- save value in temp
- reset top to top.next()
- return temp

peek (int e)

- return top value

Size

- loop through the nexts until the end.

top == null

hit Node walker = top;

at count = 0;

while (walker != null) {

Count ++;

walker = walker.getNext();

}

return Count;

GenoEKS

Object

stuff

Stack <E> {

public class ArrayStack <E> implements Stack <E> {

public E peek();

}

Stack <Integer> S =
new ArrayStack <Integer>();