Linked List

Saturday, May 28, 2022 12:19 PM

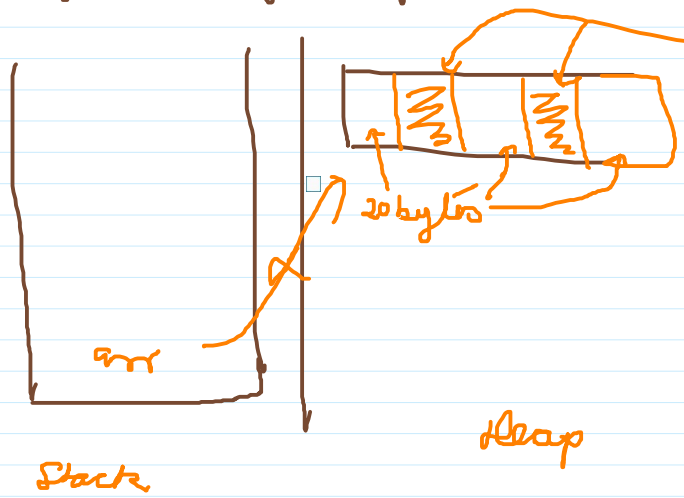
1. Array : int arr[] = new int[n]
2. ArrayList : ArrayList<Integer> arr = new ArrayList();
3. Stack: Stack<Integer> st = new Stack<>();
4. Queue : Queue <Integer> q = new ArrayDeque<>();

} Continuous memory allocation

List, there be an array

int arr = new int[6] → size = 4 × 6 = 24 bytes

Now, memory diagram :-



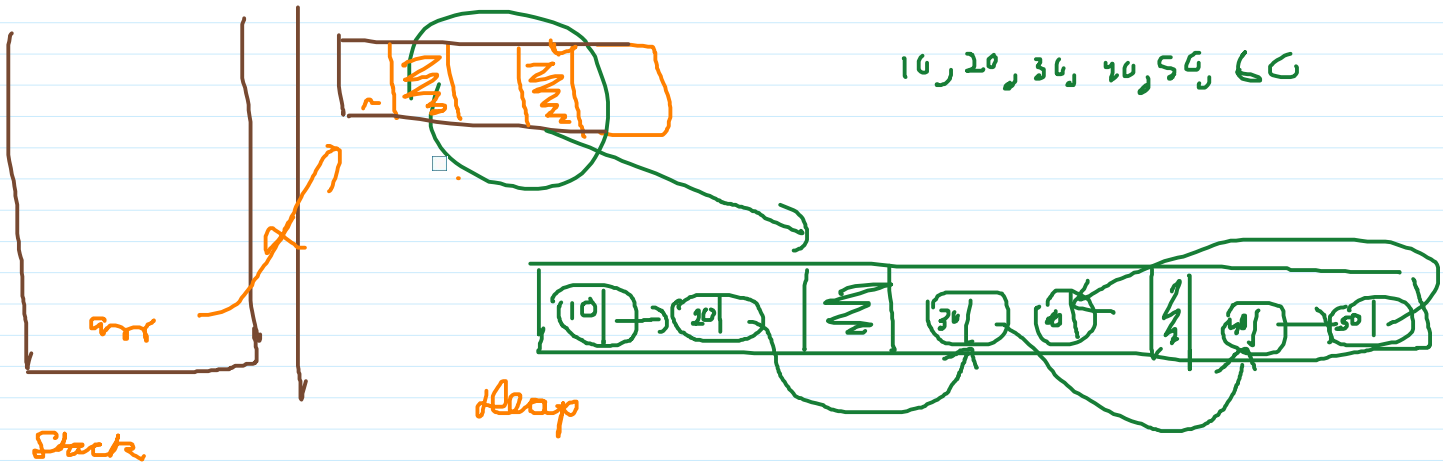
No continuous 24 bytes available even if there is 60 bytes

∴ Linked List is used

Linked List

Contains → data — 4 bytes

→ address of another node — 4 bytes — 8 bytes



→ Linked List consumes extra space but it is able to use the size in discontinuous manner

```
class Node {
    int data; → 4bytes
    Node next; → 4bytes (size depends on compiler and language)
}
```

Null point exception

→ while solving Linked List always consider following test cases

Edge test cases:

I) 0 length
II) 1 "
III) 2 "
IV) Even "
V) Odd "

if you keep this in mind then you will not face any null pt. exception