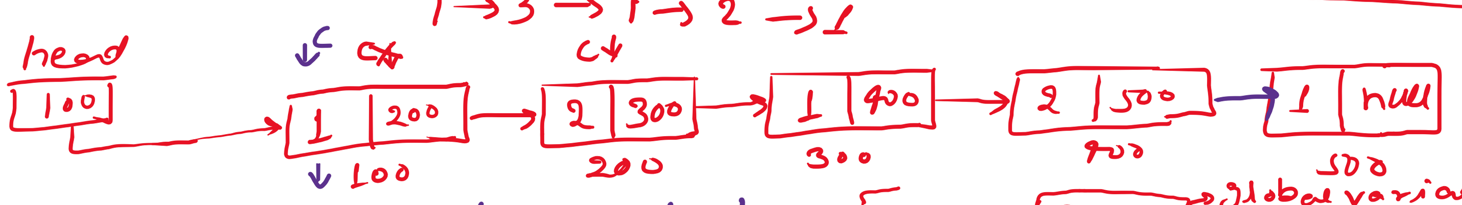
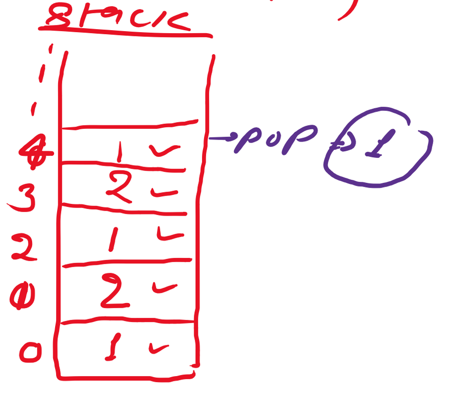
Check weather LinkedList is Palindrome or not?

use collection → Stack

1 → 3 → 1 → 2 → 1



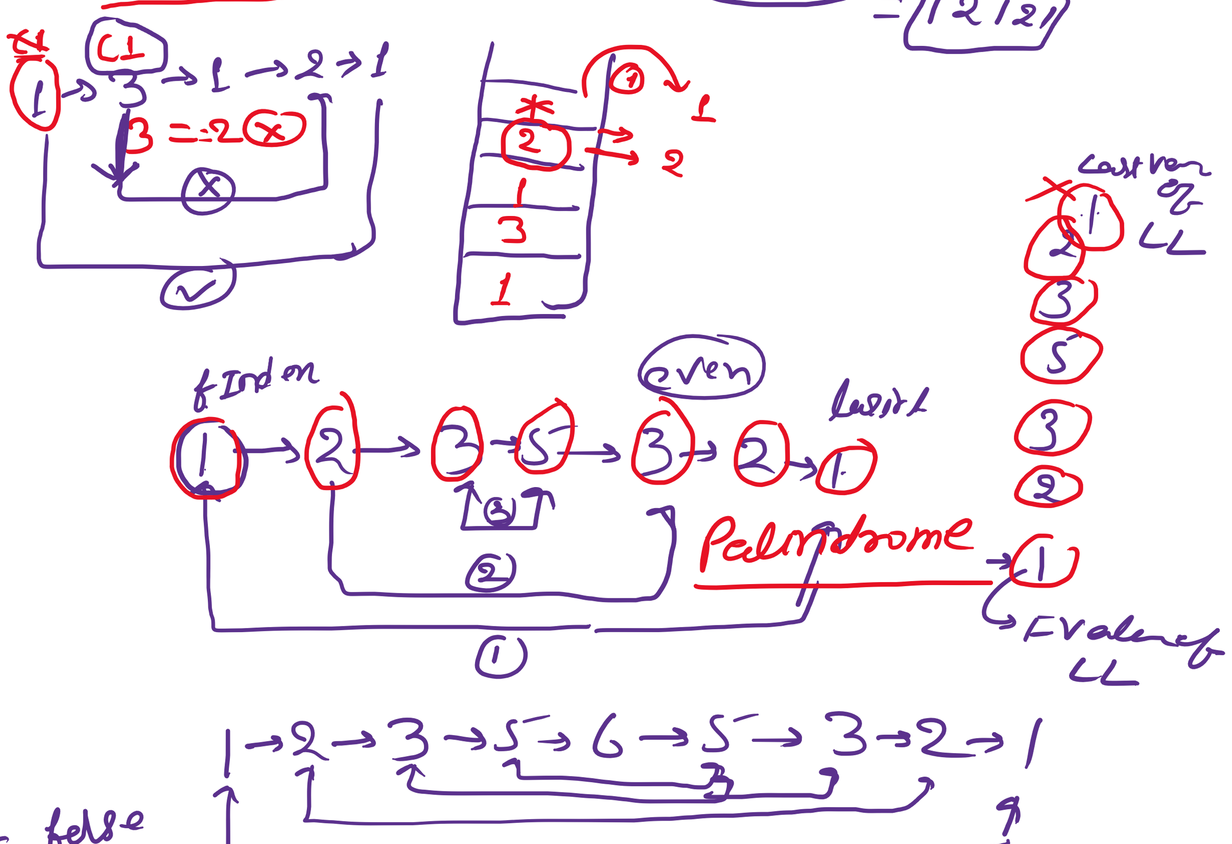
Stack(LIFO)



```
void checkPalindromeList(){
    if(head != null){
        Node current = head;
        pushData();   → boolean flag = false;
        while(current != null){
            int sdata = stac.pop();
            if(sdata == current.data){
                flag = true;
            }else{
                flag = false;
                break;
            }
            current = current.next;
        }
        if(flag){
            Sop("Palindrome");
        }else{
            Sop("not Palindrome");
        }
    }
}
```

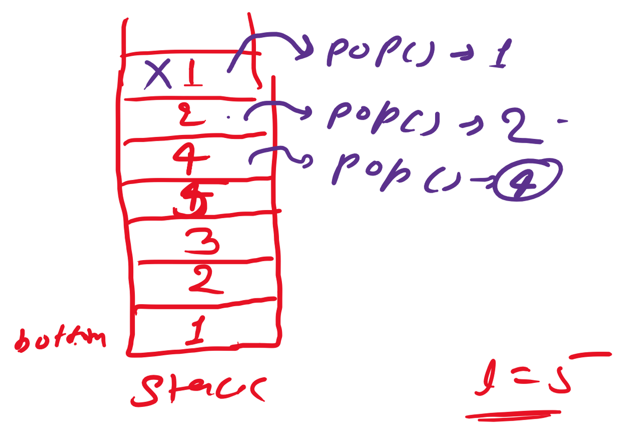
```
Stack stack = new Stack();   → global variable
void pushData(){
    if(head != null){
        Node c1 = head;
        while(c1 != null){
            stac.push(c1.data);
            c1 = c1.next;
        }
```

1 2 1 2 → reverse = 1 2 1 2 1



f = T else flag = false breaks

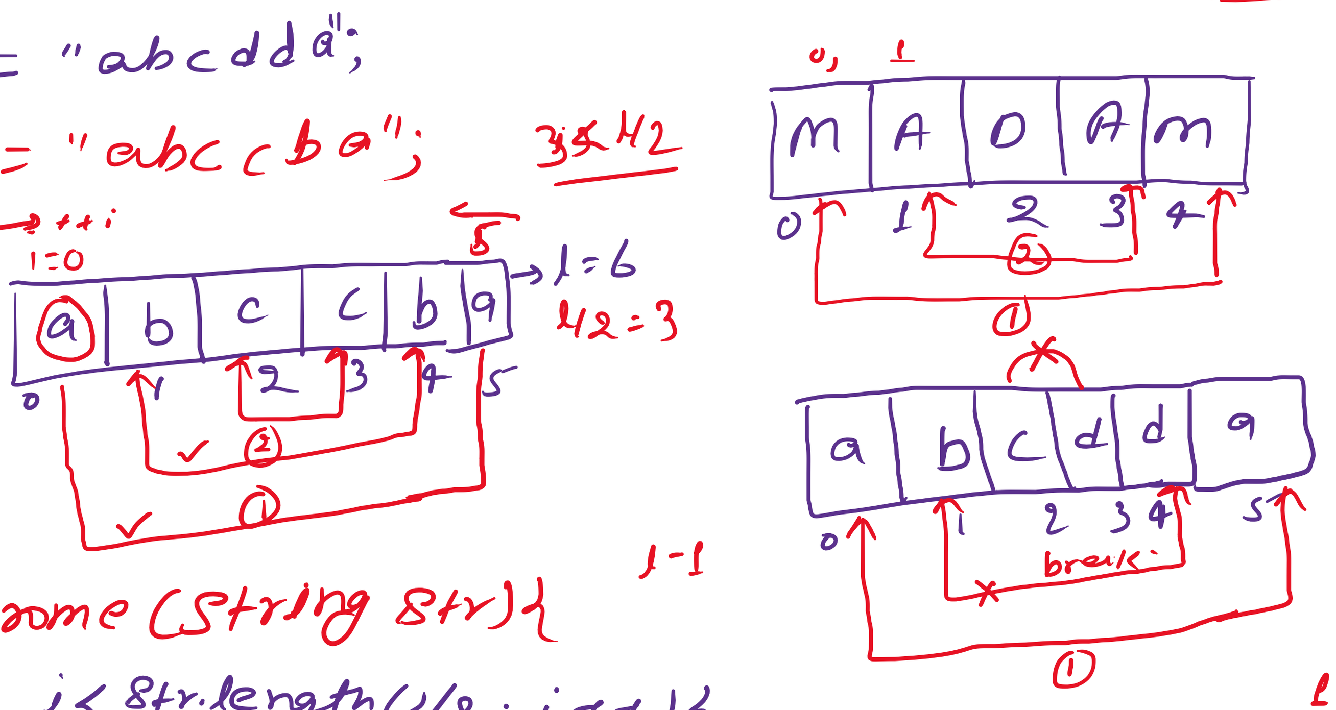
1 → 2 → 3 → 5 → 4 → 2 → 1 → not palindrome



```
String str = "MADAM"
String s2 = "abcdda";
String s3 = "abccba";
```

l = 5, l/2 = 2 (0,1)

l = 6, l/2 = 3



```
void checkPalindrome(String str){
    for(int i = 0; i < str.length()/2; i++){
        boolean flag = false;
        if(str.charAt(i) == str.charAt(str.length() - 1 - i)){
            flag = true;
        }else{
            flag = false;
            break;
        }
    }
    if(flag){
        Sop("Palindrome");
    }else{
        Sop("not Palindrome");
    }
}
```

5 - i = 4

Reverse String

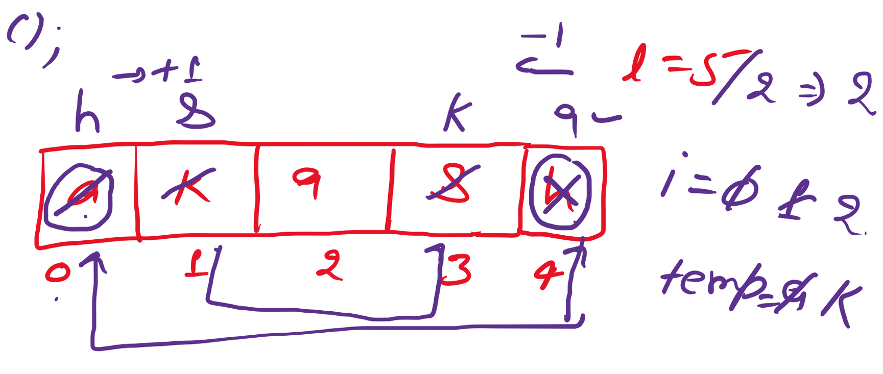
```
String str = "akash";
    char ch[] = str.toCharArray();
for(i = 0; i < ch.length/2; i++){
    char temp = ch[i];
    ch[i] = ch[ch.length - 1 - i];
    ch[ch.length - 1 - i] = temp;
}
Sop(ch);
```

(5 - 1 - 1) ⇒ 3

l = 5/2 ⇒ 2

i = 0, 1, 2

temp = a, k



h s a k a

s + str.charAt(i)

l = 1

1 ⇄ 2 ⇄ 3 ⇄ 2 ⇄ 1 → null