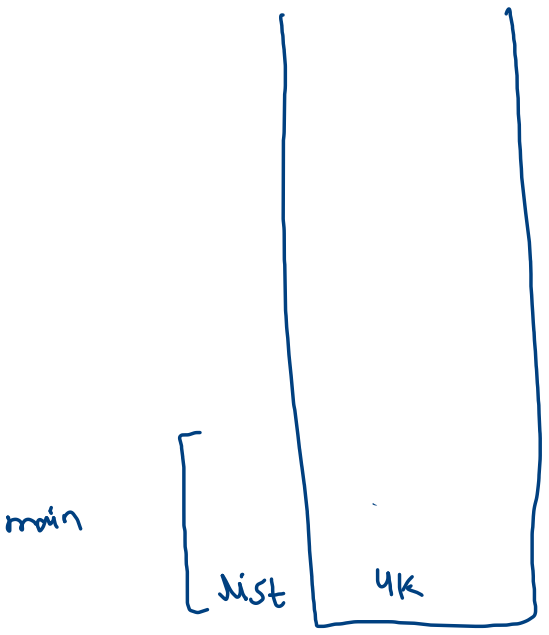
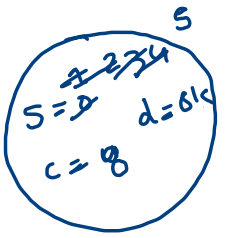
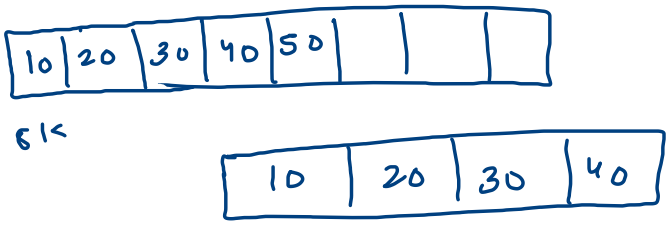
ArrayList < Integer > list = new AL < > ( );







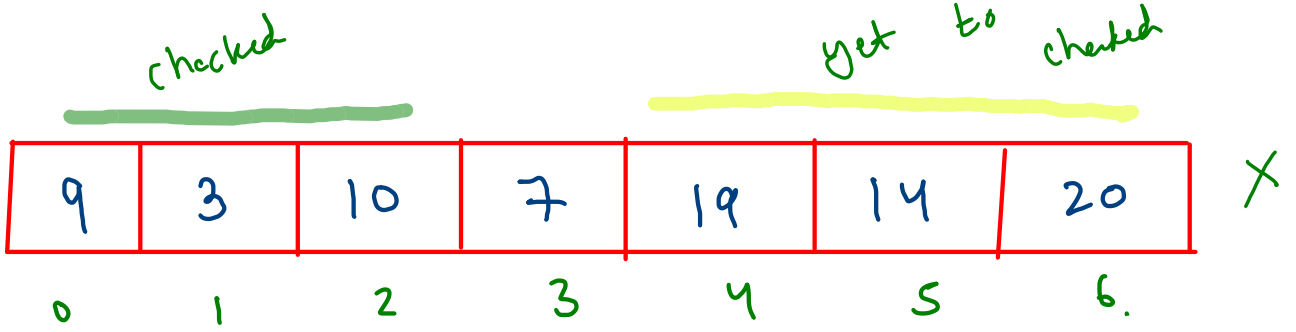
list.add(10)

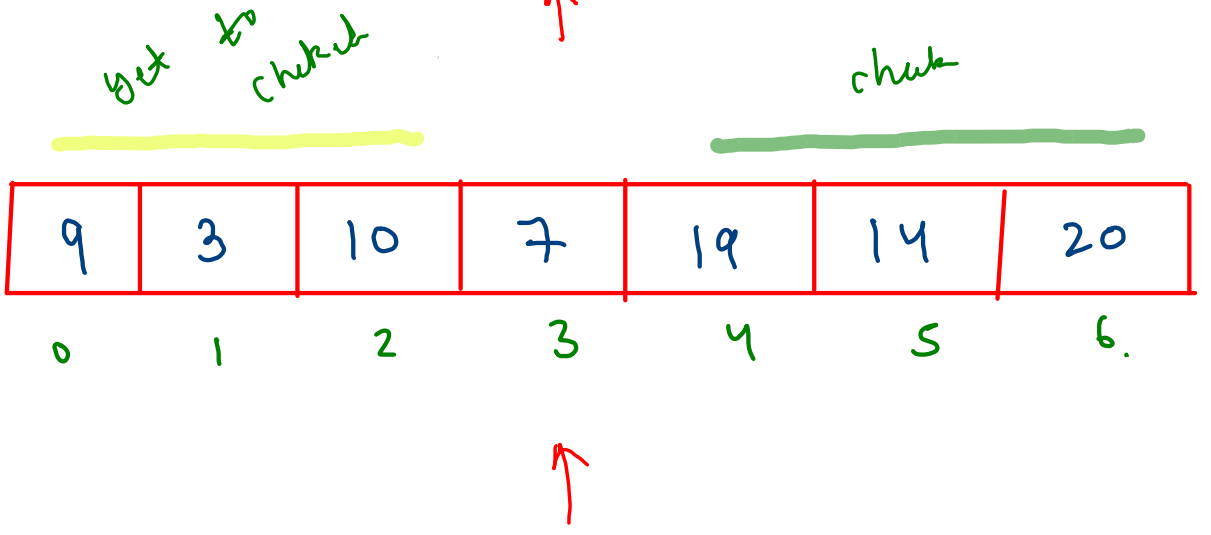
list.add(20)

list.add(30)

list.add(40)

list.add(50)



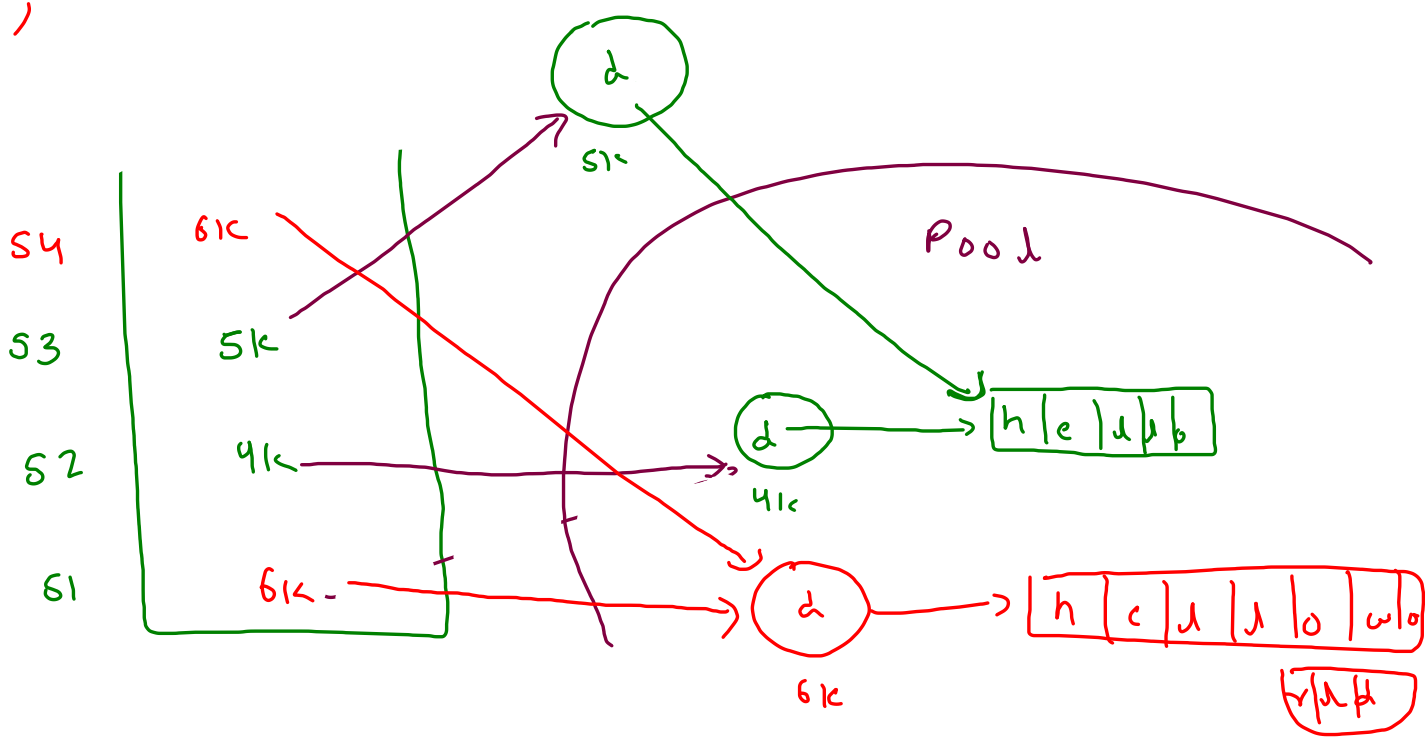



```java
public static void solution(ArrayList<Integer> al){
    // write your code here

    for(int i=0; i < al.size();i++) {
        int val = al.get(i);
        if(isPrime(val) == true) {
            al.remove(i);
        }
    }
}
```

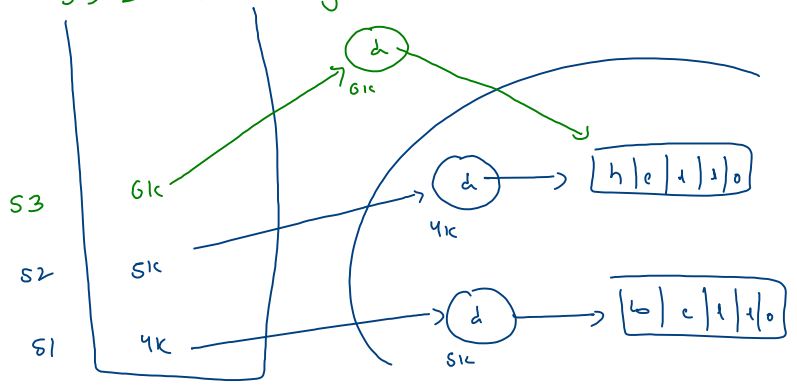
Strings

```
String s1 = "hello";
String s2 = "hello";
String s3 = new String("hello");
```

```
s1 = s1 + "hello";
String s4 = s1;
```



```
String s1 = "hello";
String s2 = "hello";
String s3 = new String("hello");
```



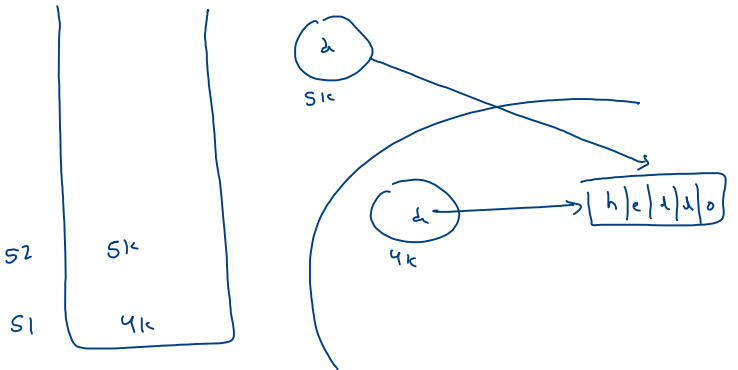

```
String s1 = "hello";
String s2 = new String("hello");
```

s1 == s2

✗ false

✓ s1.equals(s2)

true



print all palindromic substrings

a b c c b d

a

ab

abc

abcc

abccb

abccbd

b

bc

bcc

bccb

bccbd

c

cc

ccb

ccbd

c

cb

cbd

b

bd

d

for(st){

for(et: st to n-1){

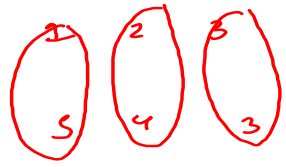
ss = s.substring(st,et+1);

if(pal(ss){

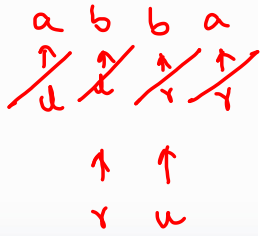
} }

abccba

↑ ↑



```java
public static void solution(String str){
    //write your code here

    for(int st = 0; st < str.length();st++) {
        for(int et = st ; et < str.length();et++) {
            String ss = str.substring(st,et+1);

            if(isPalindrome(ss) == true) {
                System.out.println(ss);
            }
        }
    }
}

public static boolean isPalindrome(String str) {
    int l = 0;
    int r = str.length() - 1;

    while(l <= r) {
        char lch  = str.charAt(l);
        char rch  = str.charAt(r);

        if(lch != rch) {
            return false;
        }

        l++;
        r--;
    }

    return true;
}
```



string = abbad

0 1 2 3 4

st -> 0, et -> 0, 1, 2, 3, 4

a ab abb abba abbad

st -> 1, et -> 1, 2, 3, 4

b bb bba bbad

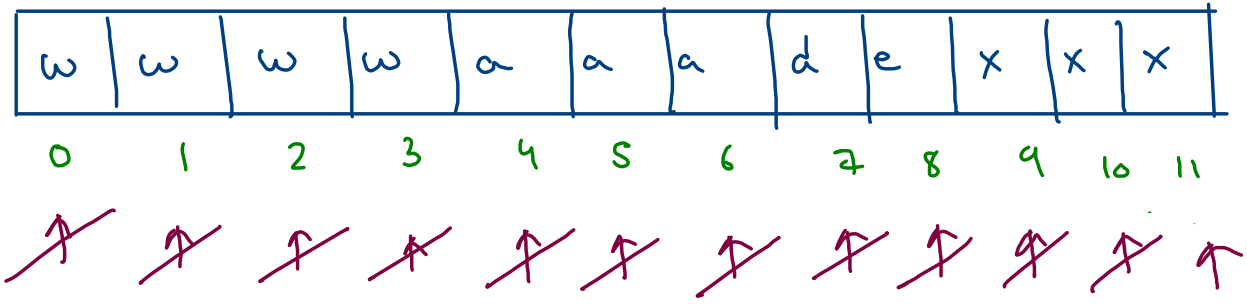
a b b a d

abba bb

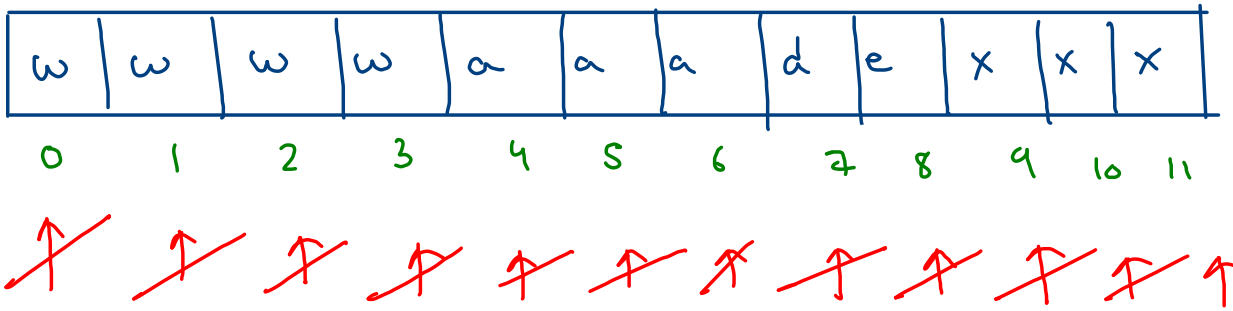
wwwwaaadexxxxxx

c1 → w a d e x

c2 → w4a3d e x 6



ans = wadex



c = ~~1~~ ~~2~~ 3

ans : w4a3dex3

```java
public static String compression2(String str){
    // write your code here

    String c2 = "";

    int c = 1;

    for(int i=0; i < str.length()-1;i++) {
        if(str.charAt(i) != str.charAt(i+1)) {
            c2 += str.charAt(i);
            if(c > 1) {
                c2 += c;
            }
            c = 1;
        }
        else {
            c++;
        }
    }

    int li = str.length()-1;
    c2 += str.charAt(li);
    if(c > 1) {
        c2 += c;
    }

    return c2;
}
```

a a b c c c

0 1 2 3 4 5

li = 5

c2 : a2bc3

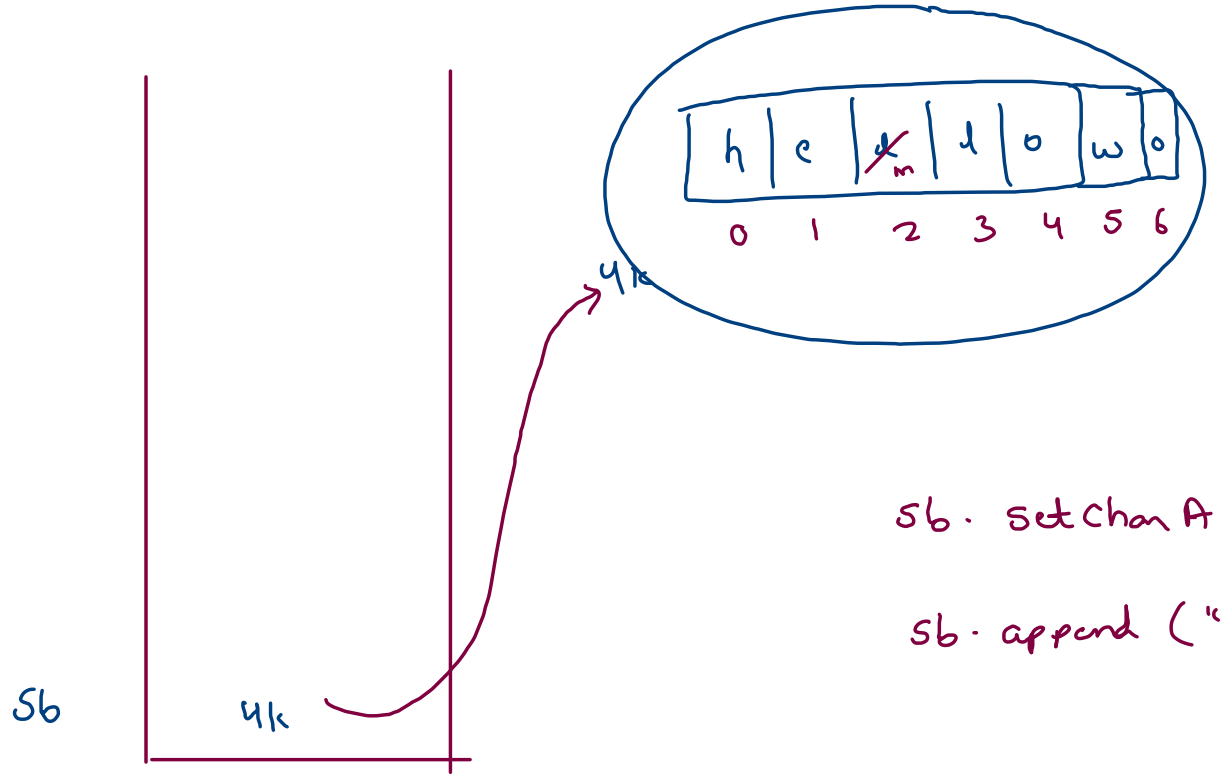
a2bc3

c = ~~1~~ ~~2~~

~~1~~ ~~1~~ ~~2~~

3

StringBuilder sb = new StringBuilder("hello");



-> mutable

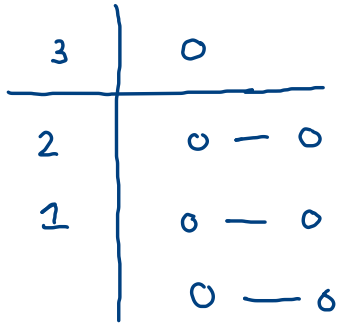
-> optimised

sb.setCharAt(2, 'm');

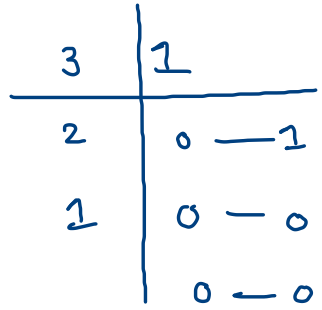
sb.append("wo");

$n = 3, \quad perm = 3! = 6$

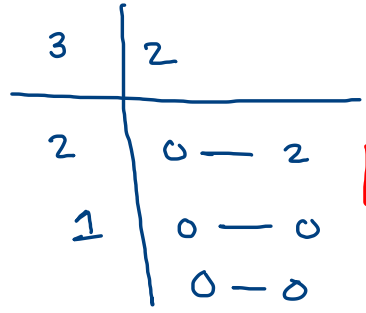
abc
bac
cab
acb
bca
cba



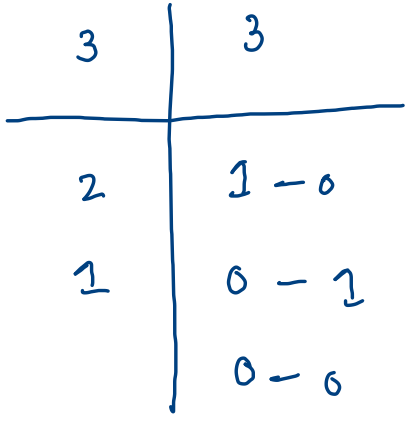
abc



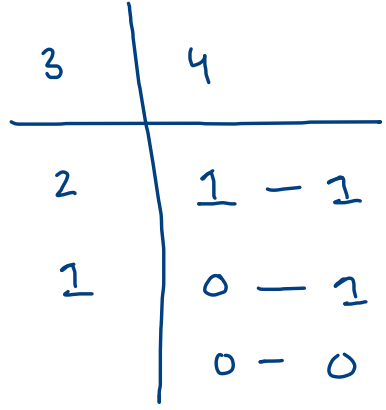
bac



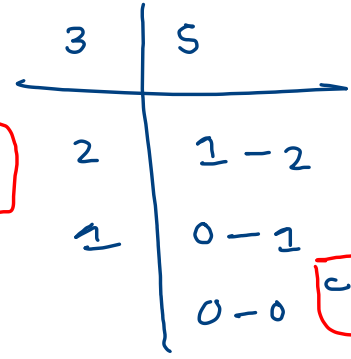
cab



acb



bca



cba

```java
int len = str.length();
int tperm = factorial(len);

for(int num = 0; num < tperm;num++) {

    StringBuilder ostr = new StringBuilder(str);
    StringBuilder perm = new StringBuilder("");
    int temp = num;

    for(int div = len; div >= 1;div--) {
        int rem = temp % div;
        temp = temp / div;

        char ch = ostr.charAt(rem);
        ostr.deleteCharAt(rem);
        perm.append(ch);
    }

    System.out.println(perm);
}
```

len = 3 tp = 6

num = 1

ostr =

temp = ~~1~~ 0

perm = bac

rem = 0

abc

bac

3	1
2	0
1	0