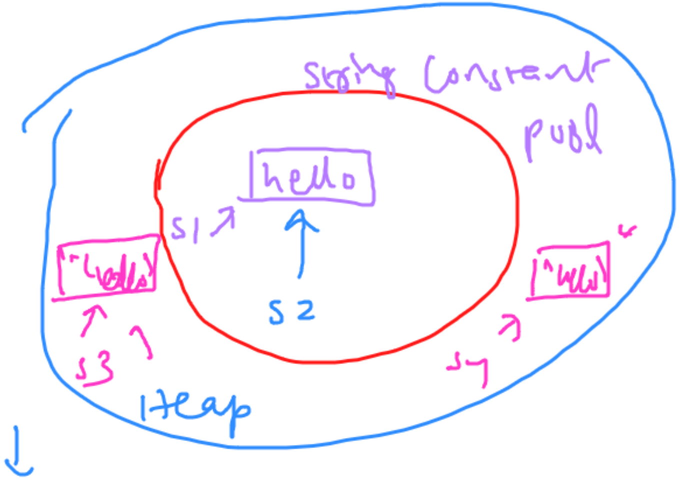
String s1 = "hello"; ✓

String s2 = "hello"; ✓



✓ String s3 = new String("hello");

✓ string s4 = new string ("hello");

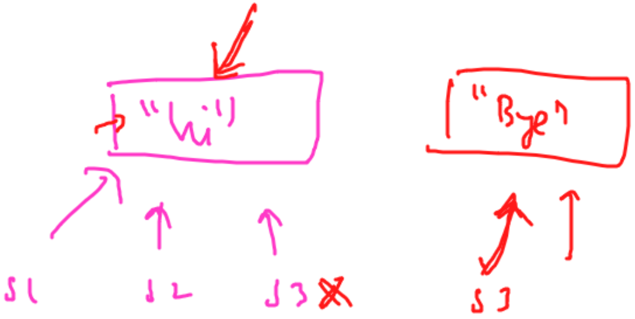
s1 == s2 → true ✓

s3 == s4 → false

→ String Immutability :-

```
string s1 = "hi";
string s2 = "hi";
string s3 = "hi";

s3 = "Bye"; ←
```

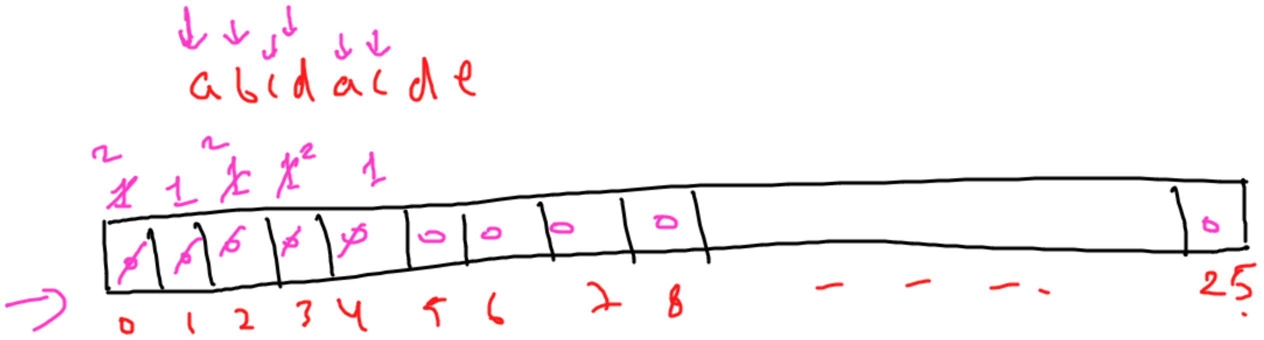


Problems :-

1. Frequency of each character of a string.
2. Find no. of unique characters in a string.
3. Check whether the string is Pangram.

"a quick brown fox jumps over the lazy dog"

Pangram → string which contains all the alphabets



2 1 2 2 1

a → 2
b → 1
c → 2
d → 2
e → 1

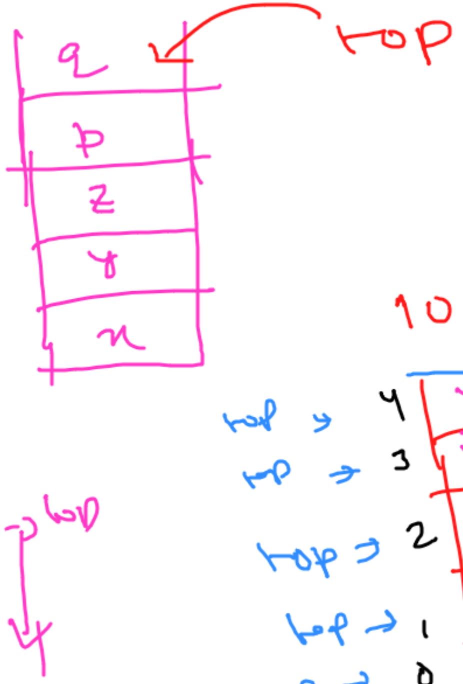
a → 0
b → 1
c → 2
d → 3
e → 4
f → 5
⋮
z → 25

Stack & Queue ← ADT

LIFO (Stack)

FIFO (Queue)

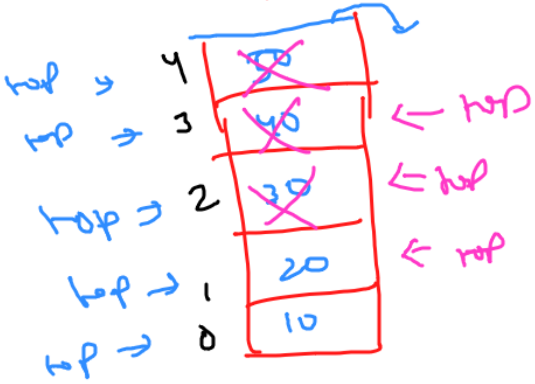
x, y, z, p, q



Insert → push

Delete → pop

10, 20, 30, 40, 50



top = ~~1~~ ~~0~~ ~~1~~ 2

top

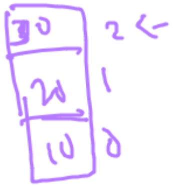
~~4~~ ~~3~~ ~~2~~ 1

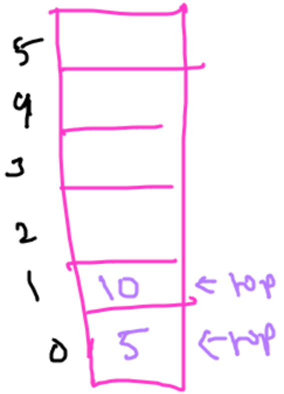
```
int stack [ ] , top = -1 , size

  int stack = new int [size];

                              <- 5
void push ( int x )
{
      if ( top == size-1)
      {
          s.o.p ("overfull");
          return;
      }
      top++;
      stack[top] = x;          stack[++top] = x;
}
```

3 = 2





top

~~-1~~ ~~0~~ 1