Date 9 Nov 2023 Page

Day-53

String Interview Questions - 3

* **Sort vowel in a string:**

=> We have to sort the vowel and put the sorted vowels in place of vowels.

Ex: "leetcode"

vowels = eeoe => eeeo

=> leetecedo ⟶ ans

=> We will have both small & large case letters.

Ex: lEetcOde

=> EeOe => EOee

ans → lEOtcede

=> Here sorting is based on the ASCII values of the characters.

Code

```
vector<int> lower (26,0);
vector<int> upper (26,0);
for(i=0; i<s.size(); i++){
    if (s[i] =='a' || s[i] =='e' || s[i] =='i' ||
  s[i]=='o' || s[i] == 'u' ){
      lower[s[i]-'a'] ++; s[i] ='#';
    else if (s[i] == 'A' || s[i] ==' E' ||... ){
         upper[s[i] - 'A'] ++; s[i] = '#';}

}string ans;
 for (i=0; i<26; i++){            }
     char c = 'A'+i;              }  O(N)
     while ( upper[i] ){          }
         ans +=c;
 }   }   upper[i]--;
```

```
for(i=0; i<26; i++){
        char c=  'a'+i;                    O(N)
        while (lower[i]){
            ans +=c;
            lower[i]--;
int first=0 ,second=0;
while( second <ans.size() ){
        if(s[first] =='#'){
            s[first] = ans[second ];
            second++;
        }
        first++;
}
return s;
```

=) If we have very very large number then we store that no. in string.

=\ So, every digit is stored as a character of 1 byte.

* ADD string

num 1 = " 26583"

num 2 = " 698 "

=) We have add two no. that are in strings.

=) We will start from end of both strings.

```
   1   1   1   ↓
0  1   2   3   4
2  6  ~~3~~5  8   3      ('3' - '0') + ('8' - '0')
       +   6   9   8          3 + 8
2  7   2   8   1               11
```

Code (num1 > num2)

```
int index1 = num1.size() - 1, index2 = num2.size()-1;
string ans;
int carry = 0 ~~;~~, sum = 0;
while( index2 >= 0){
      sum = (num1[index1] - '0') +
            (num2[index2] - '0') ~~)~~+ carry;
      carry = sum/10;
      char c = '0' + sum%10;
      ans += c;
      index2--, index1--;
}
while( index1 >= 0){
      sum = num1[index] - '0' + carry;
      carry = sum/10;
      char c = '0'+ sum%10;
      ans += c;
      index1 --;
}
if (carry)
      ans += '1';
reverse (ans.begin(), ans.end());
return ans;
```

n then

haracter

strings.

gs.