1. Find substring using z algorithm
    a. https://leetcode.com/problems/implement-strstr/

```
void createzarray(string s, int z[])
{
    int n = s.length();
    int l = 0;
    int r = 0;
    z[0] = 1;
    for(int k=1;k<n;k++)
    {
        if(k > r)
        {
            l = r = k;
            while(s[r] == s[r-l])
                r++;
            r--;
            z[k] = r - l + 1;
        }
        else
        {
            if(k + z[k-l] <= r)
                z[k] = z[k-l];
            else
            {
                l = k;
                while(s[r] == s[r-l])
                    r++;
                r--;
                z[k] = r - l + 1;
            }
        }
    }

}
```

2. https://practice.geeksforgeeks.org/problems/count-occurences-of-anagrams/0
3. Repeated substring pattern
    a. https://leetcode.com/problems/repeated-substring-pattern/submissions/
    b. If any string is periodic then there exists a substring by repeating which we can build a given string.

    A string is periodic if by right shifting k char we can rebuild the same string. → k is a length of repeating substring.

    Eg. ababab → right shift 2 times

    Abcabc → right shift 3 times

    To check all possible right shifts of a string check whether s is present in (s + s)[1:len-1]. The index is the k length of rep substring.

    012345 6789

    ababab → abab|abab k = 2
4. https://leetcode.com/problems/decoded-string-at-index/submissions/
5. https://leetcode.com/problems/decode-string/