Explanation -
Rotation of string
Abcdef

cdefab

Subsequence -
Abcdefghij
Cgh
Afi

{}	1	nC0
A b c	n	nC1
Ab ac ad ae ,,... bc bd be bj..	_ _	nC2
Abc,		
....		
Abcdefghij	1	nCn
	total	2^n

Substring
Abcdefghij

Cfg not a substring

Cde
Abc
Abcdef

A b c d e f		n
Ab bc cd de ef		n-1
Abc bcd cde def		n-2
Abcdefghi	bcdefghij	2
Abcdefghij		1
	Total =	n(n+1)/2

Permutations

Abcdabcd

Aabbccdd
Aabbcddc

Aabbcdcd

Find next permutation of this number
124631

Anagrams of string

Explanation -
Longest Palindrome substring in a string
Brute force - check all substrings time- O(n*n*n) space- O(1)

abcdcfgh
Go to every index -
Odd length - consider it as center and j=i-1,k=i+1 and j--,k++ check s[j]==s[k]
Even length - j=i,k=i+1, and j--,k++, check s[j]==s[k]
Time - O(n*n)

Longest common prefix
Abcdef
Abcfde
Abc
Ab

Ans = ab
Time - O(n*k)

Pattern matching
String - aaaabcdaaddccbfdf
Pattern - abcd

Basic brute - O(n*k)
Kmp
Rabin karp
Boyer Moore Algorithm
Just read and understand for now

GREEDY

Fractional knapsack

Bag - 11	W
Weights - 2, 5, 7, 8	w1, w2, w3, w4
Values - 1, 4, 2, 3	v1, v2, v3, v4

Find max value

Find value/weight

½	⅘	2/7	⅜	- value for 1 unit of weight

Take max

11		
⅘	11-5=6	4
½	6-2=4	4+1=5
2/7	4 - 4 = 0	5 + (4*2/7)

Min number of flips

`0001010111`

0001010111
0101010101
1010101010

How to find subsequences of string/array

Subsequence - 2^n

A b c d   b, b d , a d, a b c d, ()

_ _ _ _		
0 0 0 0	0	{}
0 0 0 1	1	a
0 0 1 0	2	b
0 0 1 1	3	a b
0 1 0 0		c
0 1 0 1		a c
0 1 1 0		b c
…		
1 1 1 0		b c d
1 1 1 1	2^n -1	a b c d

Code -

Array n elements

```
Int m = ( 1<<n);
for(int i=0;i<m;i++)
{
        int x = i;
        for(int j=0;j<n;j++)
        {
                if((x&(1<<j)) >0)
                {
                        cout<<a[j]<<" ";
                }
        }
        cout<<endl;
}
```

Time complexity - O(n* 2^n)

0 1 0 0

j=0	0 0 0 1	0
j=1	0 0 1 0	0
j=2	0 1 0 0	1
j=3	1 0 0 0	0