Contents

1 DP

1.1 LCS

```cpp
#include <bits/stdc++.h>
#define IOS
    ios_base::sync_with_stdio(false);cin.tie(0);cout.tie(0)
using namespace std;
string s1, s2;
int dp[505][505];
int main(){
    IOS
    cin >> s1 >> s2;
    memset(dp, 0, sizeof(dp));
    int l1 = s1.size(), l2 = s2.size();
    for(int i = 1;i <= l1;i++){
        for(int j = 1;j <= l2;j++){
            if(s1[i - 1] == s2[j - 1]) dp[i][j] =
                dp[i - 1][j - 1] + 1;
            else dp[i][j] = max(dp[i - 1][j], dp[i][j
                - 1]);
        }
    }
    cout << dp[l1][l2] << '\n';

    return 0;
}
```

1.2 LIS $O(n^2)$

```cpp
#include <bits/stdc++.h>
#define IOS
    ios_base::sync_with_stdio(false);cin.tie(0);cout.tie(0)
using namespace std;
typedef long long ll;
int main(){
    IOS
    int arr[100];
    int n;
    cin >> n;
    for(int i = 0;i < n;i++) cin >> arr[i];
    int dp[100];
    for(int i = 0;i < n;i++) dp[i] = 1;
    for(int i = 0;i < n;i++){
        for(int j = 0;j < i;j++){
            if(arr[i] > arr[j])
                dp[i] = max(dp[j] + 1, dp[i]);
        }
    }
    int ans = 1;
    for(int i = 0;i < n;i++) ans = max(ans, dp[i]);
    cout << ans << '\n';

    return 0;
}
```

1.3 LIS $O(n \log n)$

```cpp
class Solution {
public:
    int lengthOfLIS(vector<int>& nums) {
        vector<int> v;
        int n = nums.size();
        for(int i = 0;i < n;i++){
            int p = lower_bound(v.begin(), v.end(),
                nums[i]) - v.begin();
            if(p == v.size()) v.push_back(nums[i]);
            else v[p] = nums[i];
        }
        return v.size();
    }
};
```

1.4 LIS $O(n \log n)$

```cpp
for(int i=0;i<num.size();i++){
    if(lis.empty()||lis.back()<num[i]){
        lis.push_back(num[i]);
        dp[i]=lis.size();
    }
    else{
        auto iter=lower_bound(all(lis),num[i]);
        dp[i]=iter-lis.begin()+1;
        *iter=num[i];
    }
}
int length=lis.size();
for(int i=num.size()-1;i>=0;i--){
    if(dp[i]==length){
        ans.push_back(num[i]);
        length--;
    }
}
```

2 Prime

2.1 質數篩 CPP

```cpp
bitset<MAXN> prime_bool;
vector<ll> prime;
void find_prime(){
    prime_bool.set();
    for(int i=2;i<MAXN;i++){
        if(prime_bool[i]){
            prime.push_back(i);
        }
        for(auto j:prime){
            if(j*i>=MAXN)
                break;
            prime_bool[j*i]=0;
            if(i%j==0)
                break;
        }
    }
}
```

2.2 質數篩 PY

```python
is_prime = n * [1]
is_prime[0] = is_prime[1] = 0

for i in range(2, n):
    if is_prime[i]:
        for j in range(2, n):
            if i * j >= n:
```

```
                break
            is_prime[i * j] = 0
```

2.3 單一質數

```cpp
bool prime(int n){
    if(n < 2) return false;
    if(n <= 3) return true;
    if(!(n % 2) || !(n % 3)) return false;
    for(int i = 5;i * i <= n;i += 6)
        if(!(n % i) || !(n % (i + 2))) return false;
    return true;
}
```

2.4 egcd CPP

```cpp
int exgcd(int a,int b,int &x,int &y){
    if(b==0){
        x=1,y=0;
        return a;
    }
    int gcd=exgcd(b,a%b,y,x);
    y-=a/b*x;
    return gcd;
}
```

2.5 egcd PY

```python
def egcd(a: int, b: int) -> Tuple[int, int, int]:
    """return (g, x, y) such that a*x + b*y = g =
        gcd(a, b)"""
    """x % y"""
    if a == 0:
        return (b, 0, 1)
    else:
        b_div_a, b_mod_a = divmod(b, a)
        g, x, y = egcd(b_mod_a, a)
        return (g, y - b_div_a * x, x)
```