B – Yet Another String Algorithm

Chin Ka Wang {rina__owo}

2025-05-16

Background

Problem Idea by `rina__owo`

Preparation by `rina__owo`, `pepper1208`

Problem Restatement

Given a string *S* and a string *T*, find *T* in *S* using the KMP algorithm.

Output the prefix function calculated in the KMP algorithm and the position *T* first appears in *S*.

Subtasks

- Attempts: 5
- Max: 100
- First solved by **Chan Tsz Hang** at **29m 15s**

Subtasks Constraints

- Points are given per checkpoint in this problem. You can get the point of the checkpoint when you pass them.
- There are 10 checkpoints in total, each carry 10 points.

What is KMP algorithm?

- https://www.youtube.com/watch?v=af1oqpnH1vA

How to calculate the prefix function?

$$P[i] = \max_{k \in S}\left(P[k-1]+1 \text{ if } T[i] = T\left[P[k-1]\right], 0\right) \text{ where } S = \left\{i, P[i-1], P\left[P[i-1]-1\right], \cdots\right\}$$

Translating the pseudocode into C++

After computing P, the KMP algorithm can be implemented as follows:

```
subprogram KMP(S, T, P)
    N ← size of S
    M ← size of T
    i ← 0
    j ← 0
    while i < N and j < M do
        if S[i] = T[j] then
            i ← i + 1
            j ← j + 1
        else if j = 0 then
            i ← i + 1
        else then
            j ← P[j-1]
    if j = M then
        return i - j
    return -1              // Not found
```

Translating the pseudocode into C++

- Expected score: **100 AC!**

Takeaways

- Be familiar with different math symbols.
- Learn how to understand pseudocode. Algorithms will be often written in pseudocode in problems.
- Make great use of the samples given to let you understand more about the problem.
- Try actually substitute different numbers into the formula to figure out its pattern. This may help you understand it.