

BTVN

Cho văn bản P: “ tôi thay bạn thân mình ở hiện tại có nhiều điểm khác so với quá khứ ngay ngày ông bà ông nghech trước kia tôi đã khôn lớn thật rồi”

Mẫu T:”tôi”

Yêu cầu:

- 1) Cài đặt 2 thuật toán Rabin Karp và KMP xác định vị trí của mẫu T so với văn bản P.
- 2) Nhận xét về 2 thuật toán

Bài làm

- 1) *Cài đặt thuật toán Rabin Karp:

```
void rabin_karp(string& a, string& b, int x)
{
    int n = a.length();
    int m = b.length();
    int h = 1, d = 26;
    for (int i = 0; i < m - 1; i++)
    {
        h = (h * d) % x; //h=pow(d,M-1)
    }
    int h1=0, h2 = 0;
    for (int i = 0; i < m; i++)
    {
        h1 = (d * h1 + a[i]) % x; //h1: giá trị hash của chuỗi a
        h2 = (d * h2 + b[i]) % x; //h2: giá trị hash của chuỗi b
    }
    for (int i = 0; i <= n - m; i++)
    {
        if (h1 == h2) //if giá trị hash giống nhau thì tiến hành kiểm tra ký tự
            trong chuỗi a và b
        {
            int flag = 0;
            for (int j = 0; j < m; j++)
            {
                if (a[i + j] != b[j])
                {
                    flag = 1;
                    break;
                }
            }
            if (flag == 0)
            {
                cout << "Pattern found at index: " << i + 1 << endl;
            }
        }
        if (i < n - m) //xóa ký tự đầu tiên và thêm ký tự vào cuối sau đó tính
            giá trị hash tiếp theo
        {
            h1 = (d * (h1 - a[i] * h) + a[i + m]) % x;
```

```
        if (h1 < 0)
        {
            h1 = (h1 + x);
        }
    }
}
```

Sau khi chạy thấy hash của T= 24 và tìm được T tại vị trí 1 và 102

*Cài đặt thuật toán KMP

2) Nhận xét về 2 thuật toán

2.1 Thuật toán Rabin Karp

Thuật toán Rabin Karp không đi qua mọi ký tự trong giai đoạn đầu mà lọc các ký tự không khớp(hàm băm) rồi thực hiện so sánh.

Là một bài toán của đối sánh chuỗi có thể thực hiện bằng cách chuyển đổi một chuỗi ký tự sang kiểu số rồi tiến hành so sánh->tiết kiệm thời gian

Nhược điểm:

-Việc cài đặt cũng như thực hiện thuật toán phức tạp trong trường hợp xấu nhất là $O(mn)$ (trường hợp trùng khớp nhầm)

2.2 Thuật toán KMP

Giúp loại bỏ những ký tự đã so sánh trước đó so với các phương pháp thông thường-> thời gian chạy khá nhanh, độ phức tạp $O(n)$

Nhược điểm:

-Trong thực tế, thuật toán KMP làm việc không tốt đối với tìm kiếm trong văn bản ngôn ngữ tự nhiên, bởi vì nó chỉ có thể bỏ qua các ký tự khi phần đầu của từ giống với một phần trong văn bản. Trong thực tế điều này chỉ đôi khi xảy ra trong các văn bản ngôn ngữ tự nhiên.

