

数据结构

2022

实 验 报 告

实验项目名称： 单链表实现多项式乘法

班级： 7

学号： 2021302181140

姓名： 应晓宇

指导教师： 沈志东

实验时间： 2022. 4. 14

实验四： 实现顺序串的各种模式匹配算法

一、实验要求

- (1) 独立完成实验
- (2) 撰写实验报告

二、实验环境

硬件： CPU： AMD RYZEM 5900HX GPU： RTX 3070

操作系统： windows 10

软件： visual studio2022

三、实验步骤及思路

- (1) 题目分析： 题目要求建立串，因此先建立串

```
#include<stdio.h>

#define maxsize 100
typedef struct
{
    char data[maxsize];
    int length;
}str;
void strassign(str &s, char cstr[])
{
    int i;
    for (i = 0; cstr[i] != '\0'; i++)
        s.data[i] = cstr[i];
    s.length = i;
}
```

然后要求各种算法，简单匹配，kmp，kmp 改进

```

1 int easy(str s, str t)
2 {
3     int i = 0, j = 0;
4     while (i < s.length && j < t.length)
5     {
6         if (s.data[i] == t.data[j])
7         {
8             i++;
9             j++;
10        }
11        else
12        {
13            i = i - j + 1;
14            j = 0;
15        }
16    }
17    if (j ≥ t.length)
18        return (i - t.length);
19    else
20        return -1;
21 }

```

```

1 int kmp(str s, str t)
2 {
3     int next[maxsize], i = 0, j = 0;
4     getnext(t, next);
5     while (i < s.length && j < t.length)
6     {
7         if (j == -1 || s.data[i] == t.data[j])
8         {
9             i++;
10            j++;
11        }
12        else
13            j = next[j];
14    }
15    if (j ≥ t.length)
16        return (i - t.length);
17    else
18        return -1;
19 }

```

```

1 int kmpup(str s, str t)
2 {
3     int nextval[maxsize], i = 0, j = 0;
4     getnextval(t, nextval);
5     while (i < s.length && j < t.length)
6     {
7         if (j == -1 || s.data[i] == t.data[j])
8         {
9             i++;
10            j++;
11        }
12        else
13            j = nextval[j];
14    }
15    if (j ≥ t.length)
16        return (i - t.length);
17    else
18        return -1;
19 }

```

再是求出数组值

```

1 void getnext(str t, int next[])
2 {
3     int j, k;
4     j = 0; k = -1; next[0] = -1;
5     while (j < t.length - 1)
6     {
7         if (k == -1 || t.data[j] == t.data[k])
8         {
9             j++;
10            k++;
11            next[j] = k;
12        }
13        else
14            k = next[k];
15    }
16 }

```

```

void getnextval(str t, int nextval[])
{
    int j = 0, k = -1;
    nextval[0] = -1;
    while (j < t.length)
    {
        if (k == -1 || t.data[j] == t.data[k])
        {
            j++;
            k++;
            if (t.data[j] != t.data[k])
                nextval[j] = k;
            else
                nextval[j] = nextval[k];
        }
        else
            k = nextval[k];
    }
}

```

最后主函数调用并输出

```

void dispstr(str s)
{
    int i;
    if (s.length > 0)
    {
        for (i = 0; i < s.length; i++)
            printf("%c", s.data[i]);
        printf("\n");
    }
}

```

```

int main()
{
    int j;
    int next[maxsize], nextval[maxsize];
    str s, t;
    char ss[] = "abcabcdabcdeabcdefabcdefg";
    char tt[] = "abcdeabcdefab";
    strassign(s, ss);
    strassign(t, tt);
    printf("串s: "); dispstr(s);
    printf("串t: "); dispstr(t);
    printf("简单匹配算法: \n");
    printf("t在s中的位置=%d\n", easy(s, t));
    getnext(t, next);
    getnextval(t, nextval);
    printf("j      ");
    for (j = 0; j < t.length; j++)
        printf("%4d", j);
    printf("\n");
    printf("t[j]   ");
    for (j = 0; j < t.length; j++)
        printf("%4c", t.data[j]);
    printf("\n");
    printf("next    ");
    for (j = 0; j < t.length; j++)
        printf("%4d", next[j]);
    printf("\n");
    printf("nextval");
    for (j = 0; j < t.length; j++)
        printf("%4d", nextval[j]);
    printf("\n");
    printf("kmp算法: \n");
    printf("t在s中的位置=%d\n", kmp(s, t));
    printf("改进的kmp算法: \n");
    printf("t在s中的位置=%d\n", kmpup(s, t));
    return 0;
}

```

四、实验结果及分析

t在s中的位置=7

遇到的问题及解决方法：数组创建函数不能直接输入字符串，需要建立字符数组。编写时用一些 c++ 语法更方便。