**实验5：KMP模式匹配（2学时）**

**[问题描述]**

给定一个自串，要求找出某个字符串中该子串的第一次出现的位置。

**[实验目的]**

（1） 深入理解字符串模式匹配算法的特性。

（2） 掌握KMP（快速匹配算法）的求解方法。

**[实验内容及要求]**

1. 给定子串P以及待查找的字符串T。
2. 掌握匹配中出现不等时，P右移的位数和开始比较的字符位置的求解方法。

**[测试数据]**

/\*

 \* @Author: Weidows

 \* @Date: 2020-11-27 15:27:06

 \* @LastEditors: Weidows

 \* @LastEditTime: 2020-12-31 20:26:04

 \* @FilePath: \Weidows\C++\homework\ing\大二上\实验五\1.cpp

 \* @Description:

 \*/

**#include** <stdio.h>

**#include** <stdlib.h>

**#include** <string.h>

**#define** MAX 32

***int*** next[MAX] **=** {**-**999}; //定义next数组

***void*** get\_next(***char*** \**T*, ***int*** *next*[])

{

***int*** i **=** 0,

      j **=** **-**1;

*next*[i] **=** j;

**while** (i **<** (***int***)strlen(*T*))

  {

**if** (j **==** **-**1 **||** *T*[i] **==** *T*[j])

    {

      i**++**;

      j**++**;

*next*[i] **=** j;

    }

**else**

      j **=** *next*[j];

  }

}

***void*** get\_nextval(***char*** \**T*, ***int*** *next*[])

{

***int*** i **=** 0,

      j **=** **-**1;

*next*[0] **=** **-**1;

**while** (i **<** (***int***)strlen(*T*))

  {

**if** (j **==** **-**1 **||** *T*[i] **==** *T*[j])

    {

      i**++**;

      j**++**;

**if** (*T*[i] **==** *T*[j])

*next*[i] **=** *next*[j];

**else**

*next*[i] **=** j;

    }

**else**

      j **=** *next*[j];

  }

}

***int*** KMP\_index(***char*** \**S*, ***char*** \**T*, ***int*** *pos*)

{

***int*** i **=** *pos*,

      j **=** 0;

**while** ((i **<** (***int***)strlen(*S*)) **&&** (j **<** (***int***)strlen(*T*)))

  { //j < strlen(T)会出错，因为strlen返回是无符号数，j=-1时，退出了while循环.

**if** (*S*[i] **==** *T*[j] **||** (j **==** **-**1))

    {

      i**++**;

      j**++**;

    }

**else**

      j **=** next[j];

  }

**if** (j **==** strlen(*T*))

  {

**return** i **-** j;

  }

**else**

**return** 0;

}

***int*** main()

{

***char*** **\***S **=** "a15645babaaabaaaab0", //母串

**\***T **=** "aaaab0";               //子串(从母串13位到最后)

***int*** pos **=** 0;                     //当前位置

***int*** index **=** 0;                   //结果

***int*** len **=** strlen(T);             //子串长度

  get\_next(T, next);

  printf("index = %d\n", KMP\_index(S, T, pos));

**return** 0;

}

**测试使用的子母串**

***char*** **\***S **=** "a15645babaaabaaaab0", //母串

**\***T **=** "aaaab0";               //子串(从母串13位到最后)

