```
第四章串
         16:07
2020年11月10日
   Q:是字母、数字或其它字符;n为串的长度,n=ogh称为空串。
   子率:一个事中任营个连续字符组成的子序列(空空里)积为这事的子事。
   宰相等:长度且对应位置的字符者时相等.
                                 //预定义最大事长为255
        char ch[MAXLEN];
        int length;
                                //事的实际长度
    举约两种表示方法:
     (1)如此出,用一个额外变量Len来存放率的长度。
     (2)在事值台面加一个不计入事长的结束标记字符"10",此时事长为隐含值。
   2、维地有结结构:
    块链结构: #define CHUNKSIZE 80 //空外线的大小.
             typedef struct Chunk [
                char ch[CHUNKSIZE];
                  struct Chunk *nest;
                                       //事的兴鹅针和尾档针.
                  Chunk *head, *tail;
                 int curlen;
                                      //事的当前长度.
                                       / 扫将事的块链结构.
结点为中的块链
结织物的链表一个图子图子
                       (注)在生生表中,指针土建与用4个字符
三、串的模式匹西乙氧法:
   算法目的:在静定主事中的含于事(模型事)第一次出现的位置。(定位).
   算法种类: BF算法.(Brute-Force, 经典算法, 速度人量).
           KMP算法.(速度快).
   BF算法
   int Index (SString S, SString T, int pos){
                                           int i = pos, j = 1;
       while (i <= S.length 2& j <= T.length){
if(S.ch[i]==T.ch[i]){
                                           //丝缕比较下一位.
                                           //指军七台區重案/开始匹西己
        if(j>T.length)
return i-T.length;
                                           // 鱼回海第一个字符的位置。
        else
           return 0;
   BF算法最坏。时间复杂度为D(nm). n与m分别为主事和模式事的长度.
四、KMP算法:
  主宰S的指针证必回溯,可搜速到0(n+m).
  void get_next(String T, int next[]){
      int i=1, j=0;
     ne\pi t[i] = 0;
     while(i < T.length) {

if (j == 0|| T.ch[i] == T.ch[j]) {
            tti; ttj;
            next[i]=j;
            i=next[j];
   int Index_KMP (String S, String T, int next[], int pos){
      int i=pos, j=1;
      while (i \le S. length && j \le T. length) {

if (j == 0|| S. ch[i] == T. ch[i]) {

++i;++j;
                                 // 模型单向右移动。
             j==nent[j];
      if (j>T.length)
return i-T.length;
                                 // 匹西己成功.
      else
          return D;
         1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
   nextval值
              M=ax
```