KMP Algorithm

http://acm.cs.nthu.edu.tw/problem/10590
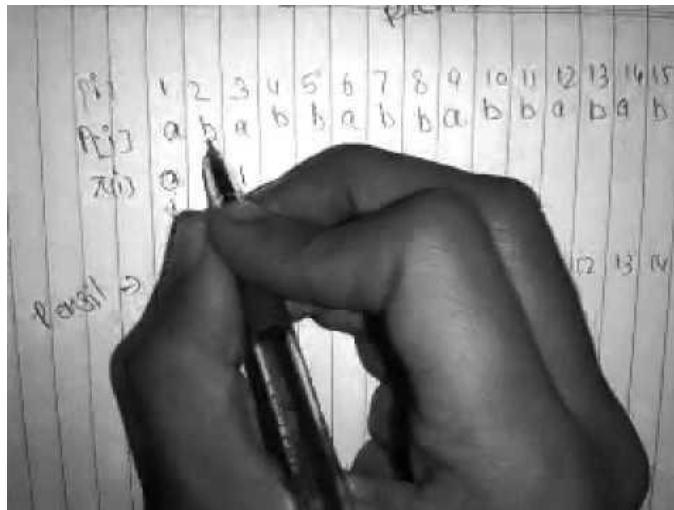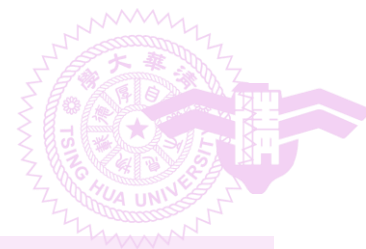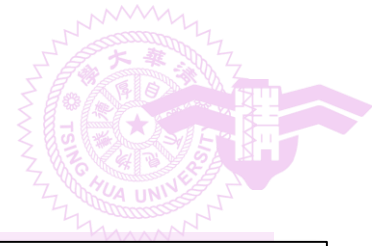
Introduction

- We want to write a program to help us performing KMP string searching

- Please refer to Section 2.6.2 for the details of the KMP algorithm
  - Program 2.16 and 2.17 are exampling code

- Some extensions
  - Our program needs to report all matched patterns instead of the first matched one
  - Our program needs to show the procedure of pattern searching

Input

2   number of string pairs

abaabaabab   pattern string
abbaaaaabaabaabaabaababababaaabaaab   a string pair
abacababaca   content string
baaabacabacabacaababababacababaca

Output

```
a b a a b a a b a b
-1 -1 0 0 1 2 3 4 5 1
a b b a a a a a b a a b a a b a b a a b a a b a b a b a a a b a a a b
= = x
    x
      = x
        = x
          = x
            = = = = = = = = = = o
                      ~ ~ = = = = = = = = o
                            ~ ~ = x
                              ~ = = = = x
                                ~ x
                                  = = = = x
                                    ~ x
                                      = =

Number of patterns found: 2

a b a c a b a b a c a
-1 -1 0 -1 0 1 2 1 2 3 4
b a a a b a c a b a c a b a b a c a a b a b a b a c a b a b a c a
x
  = x
    = x
      = = = = = = = = x
            ~ ~ ~ = = = = = = = = o
                  ~ ~ ~ ~ ~ x
                      ~ x
                        = = = x
                          ~ = = x
                            ~ = = = = = = = = = = o

Number of patterns found: 2
```

Output

pattern and the failure function

content string

```
a b a a b a a b a b
-1 -1 0 0 1 2 3 4 5 1
a b b a a a a a b a a b a a b a b a a b a a b a b a b a a a b a a a b
= = x
    x
      = x
        = x
          = x
            = x
            = = = = = = = = = o
              ~ ~ = = = = = = = = o
                    ~ ~ = x
                      ~ = = = x
                        ~ x
                        = = = = x
                          ~ x
                            = =
```
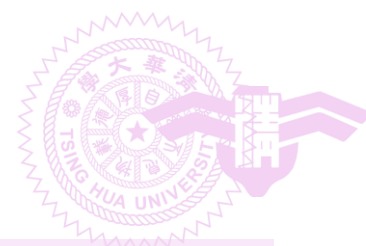
'=' denotes that the algorithm reads the corresponding character of the content string and determines that the character is a match

The corresponding character leads to a matched pattern

'~' denotes a matched character indicated by the failure function

'x' denotes a mismatch

Number of patterns found: 2

Two newlines

Procedure

Space

```
a b a c a b a b a c a
-1 -1 0 -1 0 1 2 1 2 3 4
b a a a b a c a b a c a b a b a c a a b a b a b a c a b a b a c a
x
  = x
    = x
      = = = = = = = = x
        ~ ~ ~ = = = = = = = o
            ~ ~ ~ ~ ~ x
              ~ x
                = = = x
                  ~ = = x
                    ~ = = = = = = = = = = o
```

Number of patterns found: 2