



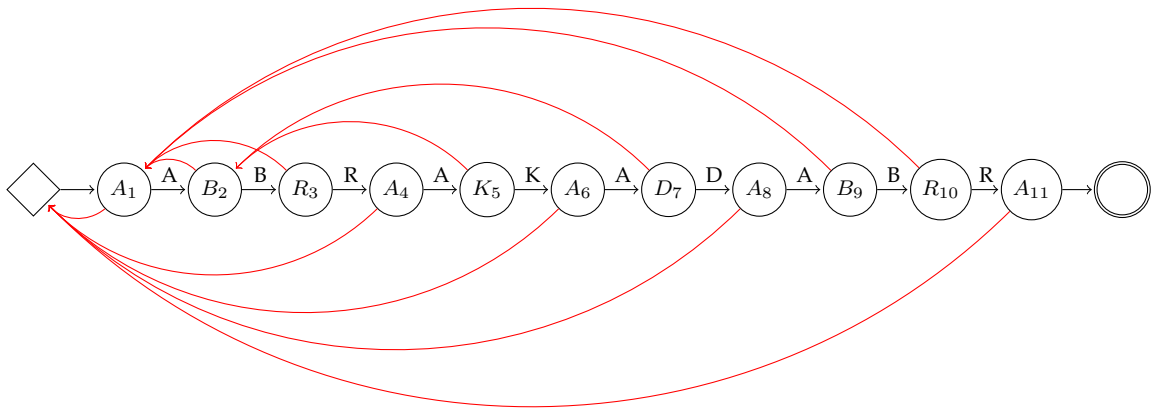
Övning 5

Automater, reguljära uttryck, syntax

1. Abrakadabra

Konstruera en KMP-automat som söker efter texten ABRAKADABRA. Ange även den next-vektor som definierar automaten.

Ungefär hur många jämförelser behövs för att automaten ska se att ordet inte finns med i "Harry Potter och Fenixorden", en bok på 1.8 MB?



i	1	2	3	4	5	6	7	8	9	10	11
$next(i)$	0	1	1	0	2	0	2	0	1	1	0

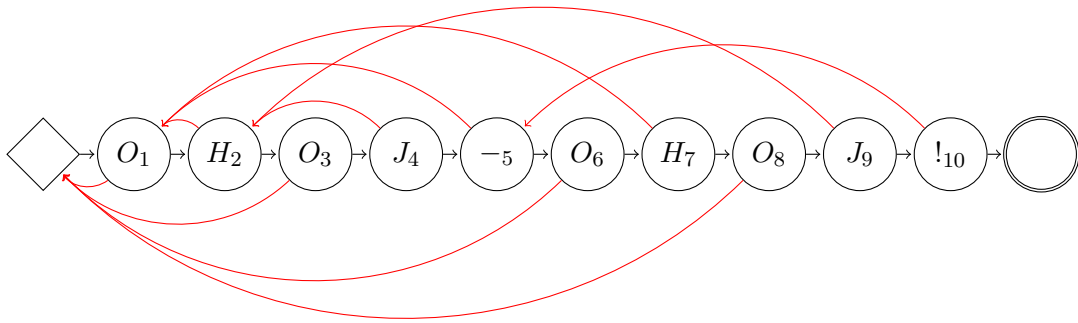
KMP-sökning tar $n + m$ jämförelser, där m är antal tecken i söksträngen och n är antal tecken i texten.

Alltså 1.8 miljoner + 11 (alt. $1.8 \cdot 2^{20} + 11 = 1887437 + 11$).

2. Värstingvrålsautomat (Tildatenta 000831)

Skriv en KMP-automat som söker efter OHOJ-OHOJ! i en textfil med nedtecknade värstingvrål från en vitmaktskonsert.

Ange också den next-vektor som definierar automaten.



i	1	2	3	4	5	6	7	8	9	10
$next(i)$	0	1	0	2	1	0	1	0	2	5

3. Regexp

a) Givet det reguljära uttrycket

$s(a|o)nd-?l\ddot{a}da$

Skriv upp tre strängar som matchas av det reguljära uttrycket och ett som inte gör det.

b) Söka efter Kronsskog. Skriv ett reguljärt uttryck som matchar alla tänkbara sätt att stava namnet Kronsskog (Crounskog, Krohnskoog, etc).

1. sandlåda, sand-låda, sondlåda men inte syndlåda

2. $[CK]rou?h?nskoo?g$

$[CK]ro[uh]?nskoo?g$ (OBS: \rightarrow Kouhuhuhuhnskog OK)

Metatecken Beskrivning

.	matchar ett tecken, vilket som helst (utom radslut)
$X?$	matchar noll eller ett X
$X+$	matchar ett eller flera X
X^*	matchar noll eller flera X
$X\{m, n\}$	matchar m tom n st X (utökad)
$[]$	matchar ett av tecknen inom klammrarna
$[^]$	matchar vilket tecken som helst förutom de inom klammrarna
$^$	matchar början på rad
$\$$	matchar slutet på en rad (men <i>inte</i> nyradstecknet därefter)
$X Y$	matchar uttrycket X eller uttrycket Y
$\backslash .$	matchar en punkt. ("escape:a metatecken")
$()$	skapar en grupp.
$\backslash n$	referera n :te gruppen (0 – 9)

Extra

$/(bb|[^\wedge]{2})/$

4. Syntax för kanadensare (Tildatenta 040313)

Olle sitter och rättar ett tentatal. Tentatalet går ut på att man ska skriva en grammatik för meddelanden av följande typ:

Kanot 42, kanot 666, kanot 4711 och kanot 17 ska in!
Kanot 1 och kanot 2 ska in!
Kanot 13 ska in!

Vilken eller vilka av följande fyra alternativ kan producera dessa meddelanden? Motivera med exempel varför de övriga inte kan producera dem.

En del av alternativen kan producera oönskade meningar, man vill t ex inte ha Kanot 1 och kanot 2, kanot 3 och kanot 4 ska in!

Vilket eller vilka av alternativen kan producera oönskade meningar? Ge exempel.

- ```
(1) <meddelande> ::= Kanot <tal><svans> |
 <meddelande> kanot <tal><svans>
<svans> ::= och| ska in! | ,
<tal> ::= 1 | 2 | 3 | ...

(2) <meddelande> ::= Kanot <tal> ska in! |
 Kanot <tal><svans>
<svans> ::= och kanot <tal> ska in! |
 , kanot <tal><svans>
<tal> ::= 1 | 2 | 3 | ...

(3) <meddelande> ::= Kanot <tal><svans>
<svans> ::= ska in! | , kanot <tal> |
 och kanot <tal>
<tal> ::= <svans> | 1 | 2 | 3 | ...

(4) <meddelande> ::= Kanot <tal><svans> |
 kanot <tal><svans>
<svans> ::= ska in! | , | och
<tal> ::= 1 | 2 | 3 | ...
```

Alternativ 1 och 2 kan producera alla meningarna.

Alternativ 3 och fyra kan inte producera

Kanot 1 och kanot 2 ska in!

Alternativ 1 och 4 godkänner felaktigt

Kanot 4 och

Alternativ 3 godkänner felaktigt

Kanot och kanot 2

## 5. Värsta webbsyntaxen (Tildatenta 000831)

En webbfil innehåller dels webbsidans text, dels taggar för radbrytningar och indragningar. Taggen `<BR>` ger ny rad och för att få indragning av ett textavsnitt skriver man taggen `<Q>` före och taggen `</Q>` efter. Exempelvis ger webbfilen

```
Organismer
 <Q> Djur

 <Q> Flugor

Sillar
 </Q> Svamp

 <Q> Flugsvamp

Sillkremla
 </Q> </Q>
```

följande webbsideutseende:

```
Organismer
 Djur
 Flugor
 Sillar
 Svamp
 Flugsvamp
 Sillkremla
```

Skriv en syntax för webbfiler där endast dessa taggar och vanlig text förekommer. Du kan få använda `<text>` för att beteckna godtycklig taggfri text.

```
<wpage> ::= <nothing>
 | <text>
 | <Q><wpage></Q>
 | <wpage>
<wpage>
<Q> ::= "<Q>"
</Q> ::= "</Q>"

 ::= "
"
```