

# Forstrengstré

Bergur Snorrason

6. apríl 2022

- ▶ Við segjum að ekki tómt rótartré  $T = (V, E)$  ásamt vörpun  $\tau: E \rightarrow \Sigma$ , þar sem  $\Sigma$  er eitthvað stafróf, sé *forstrengstré* (e. *prefix tree*).

- ▶ Við segjum að ekki tómt rótartré  $T = (V, E)$  ásamt vörpun  $\tau: E \rightarrow \Sigma$ , þar sem  $\Sigma$  er eitthvað stafróf, sé *forstrengstré* (e. *prefix tree*).
- ▶ Forstrengstré eru oft kölluð *Trie*.

- ▶ Við segjum að ekki tómt rótartré  $T = (V, E)$  ásamt vörpun  $\tau: E \rightarrow \Sigma$ , þar sem  $\Sigma$  er eitthvað stafróf, sé *forstrengstré* (e. *prefix tree*).
- ▶ Forstrengstré eru oft kölluð *Trie*.
- ▶ Við segjum að strengur  $s$  sé í trénu ef það er til hnútur  $v$  í trénu þannig að

$$s = \tau(e_1) \dots \tau(e_k)$$

þar sem  $e_1, \dots, e_k$  eru leggirnir á einfalda veginum frá rót til  $v$ , í réttri röð.

- ▶ Við segjum að ekki tómt rótartré  $T = (V, E)$  ásamt vörpun  $\tau: E \rightarrow \Sigma$ , þar sem  $\Sigma$  er eitthvað stafróf, sé *forstrengstré* (e. *prefix tree*).
- ▶ Forstrengstré eru oft kölluð *Trie*.
- ▶ Við segjum að strengur  $s$  sé í trénu ef það er til hnútur  $v$  í trénu þannig að

$$s = \tau(e_1) \dots \tau(e_k)$$

þar sem  $e_1, \dots, e_k$  eru leggirnir á einfalda veginum frá rót til  $v$ , í réttri röð.

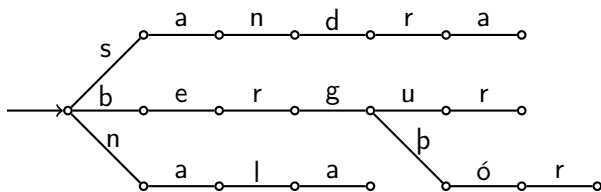
- ▶ Það er mjög algengt að geyma aukagögn í hnútunum í trénu.

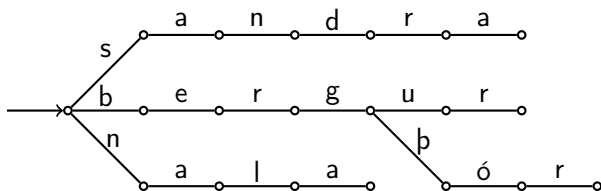
- ▶ Við segjum að ekki tómt rótartré  $T = (V, E)$  ásamt vörpun  $\tau: E \rightarrow \Sigma$ , þar sem  $\Sigma$  er eitthvað stafróf, sé *forstrengstré* (e. *prefix tree*).
- ▶ Forstrengstré eru oft kölluð *Trie*.
- ▶ Við segjum að strengur  $s$  sé í trénu ef það er til hnútur  $v$  í trénu þannig að

$$s = \tau(e_1) \dots \tau(e_k)$$

þar sem  $e_1, \dots, e_k$  eru leggirnir á einfalda veginum frá rót til  $v$ , í réttri röð.

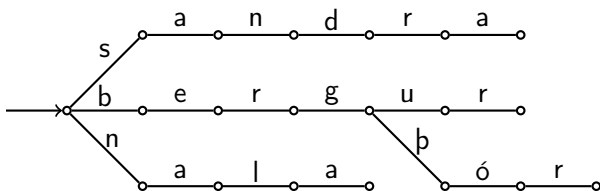
- ▶ Það er mjög algengt að geyma aukagögn í hnútunum í trénu.
- ▶ Skoðum dæmi um forstrengstré sem hefur engin aukagögn í hnútunum.





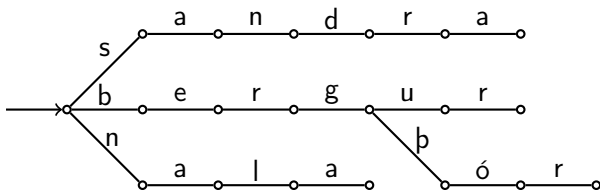
► Dæmi um strengi í trénu eru:





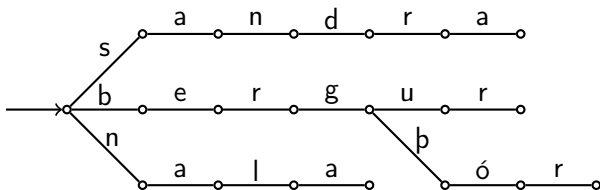
► Dæmi um strengi í trénu eru:

► „sandra”,



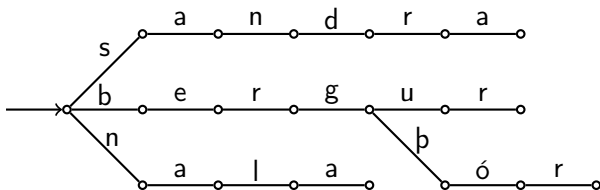
► Dæmi um strengi í trénu eru:

- „sandra”,
- „nala”,



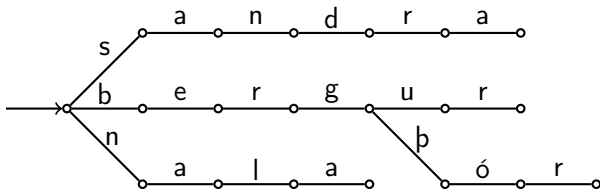
► Dæmi um strengi í trénu eru:

- „sandra”,
- „nala”,
- „bergur”,



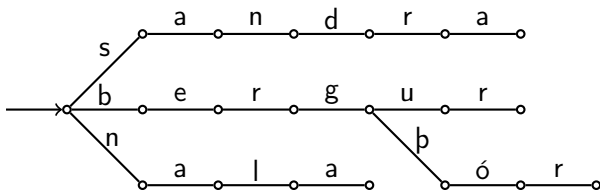
► Dæmi um strengi í trénu eru:

- „sandra”,
- „nala”,
- „bergur”,
- „bergþór”,



► Dæmi um strengi í trénu eru:

- „sandra”,
- „nala”,
- „bergur”,
- „bergpór”,
- „san” og



- ▶ Dæmi um strengi í trénu eru:

- ▶ „sandra”,
- ▶ „nala”,
- ▶ „bergur”,
- ▶ „bergþór”,
- ▶ „san” og
- ▶ „” (tómi strengurinn)

- ▶ Algengt er að merkjar suma hnúta sem *lokahnúta*.

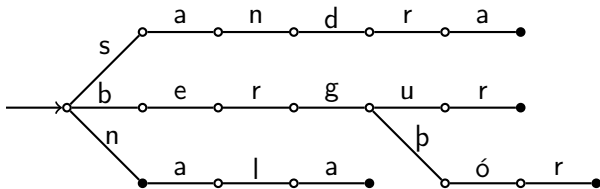
- ▶ Algengt er að merkjar suma hnúta sem *lokahnúta*.
- ▶ Þetta eru dæmi um aukagögn sem við geymum í hnútum.

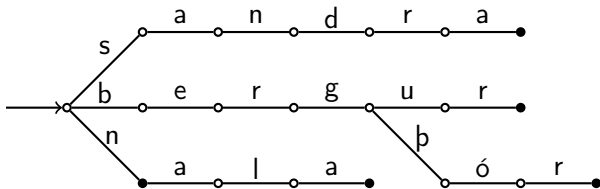


- ▶ Algengt er að merkjar suma hnúta sem *lokahnúta*.
- ▶ Þetta eru dæmi um aukagögn sem við geymum í hnútum.
- ▶ Við segjum að strengur  $s$  sé í trénu ef það er til **lokahnútur**  $v$  í trénu þannig að

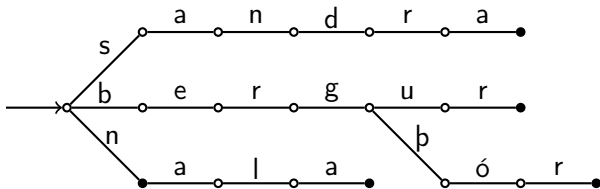
$$s = \tau(e_1) \dots \tau(e_k)$$

þar sem  $e_1, \dots, e_k$  eru leggirnir á einfalda veginum frá rót til  $v$ , í réttri röð.



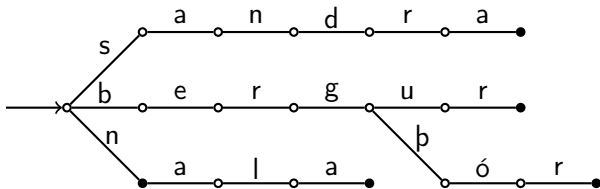


► Strengirnir í trénu eru:



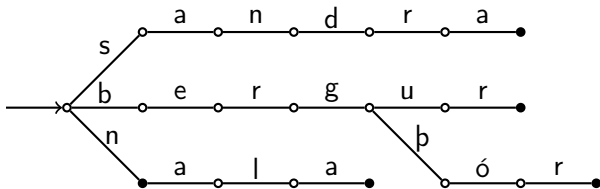
► Strengirnir í trénu eru:

► „sandra”,



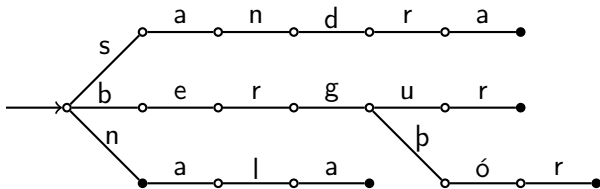
► Strengirnir í trénu eru:

- „sandra”,
- „nala”,



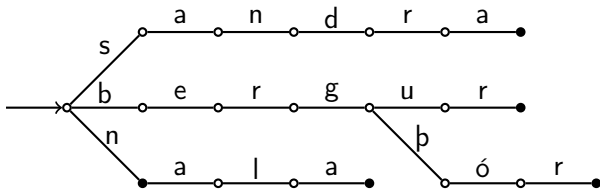
► Strengirnir í trénu eru:

- „sandra”,
- „nala”,
- „bergur”,



► Strengirnir í trénu eru:

- „sandra”,
- „nala”,
- „bergur”,
- „bergþór” og



► Strengirnir í trénu eru:

- „sandra”,
- „nala”,
- „bergur”,
- „bergþór” og
- „n”



- ▶ Hvað gerum við ef við viljum bæta streng við forstrengstré?

- ▶ Hvað gerum við ef við viljum bæta streng við forstrengstré?
- ▶ Ef við viljum setja strenginn  $s = s_1 s_2 \dots s_n$  í tréð  $T$  þá setjum við strenginn  $s' = s_2 s_3 \dots s_n$  í hluttré  $T$  sem við lendum í ef við flygjum leggnum merktum  $s_1$ .

- ▶ Hvað gerum við ef við viljum bæta streng við forstrengstré?
- ▶ Ef við viljum setja strenginn  $s = s_1 s_2 \dots s_n$  í tréð  $T$  þá setjum við strenginn  $s' = s_2 s_3 \dots s_n$  í hluttré  $T$  sem við lendum í ef við flygjum leggnum merktum  $s_1$ .
- ▶ Takið eftir að það hluttré má vera tómt (með öðrum orðum er ekki leggur merktur  $s_1$ ).

- ▶ Hvað gerum við ef við viljum bæta streng við forstrengstré?
- ▶ Ef við viljum setja strenginn  $s = s_1 s_2 \dots s_n$  í tréð  $T$  þá setjum við strenginn  $s' = s_2 s_3 \dots s_n$  í hluttré  $T$  sem við lendum í ef við flygjum leggnum merktum  $s_1$ .
- ▶ Takið eftir að það hluttré má vera tómt (með öðrum orðum er ekki leggur merktur  $s_1$ ).
- ▶ Í því tilfelli stækkar tréð.

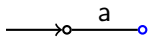


„api“

→o

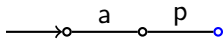
„api“



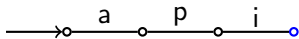


„pi“

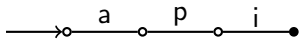




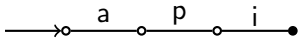
$j''$



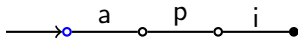
”  
”



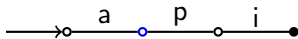
„apar”



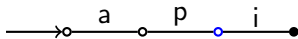
„apar”

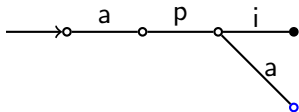


„par“



„ar“

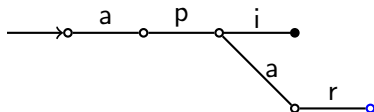


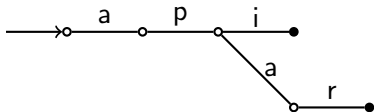


„r”

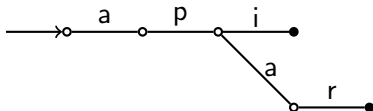


”  
”

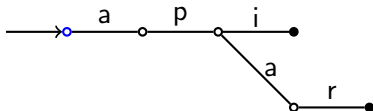




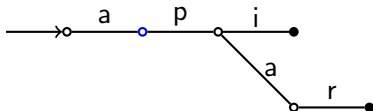
„apaköttur“



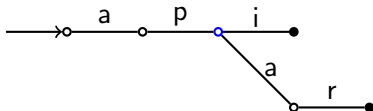
„apaköttur“



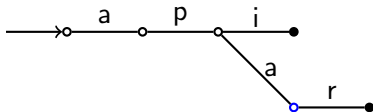
„paköttur“



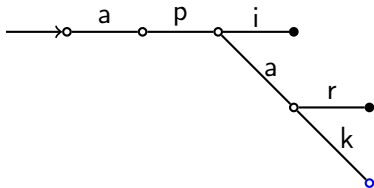
„aköttur“



„köttur“

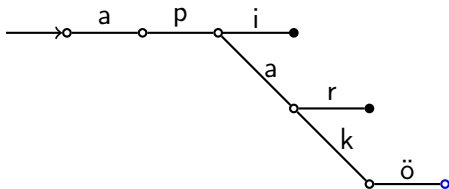


„öttur“

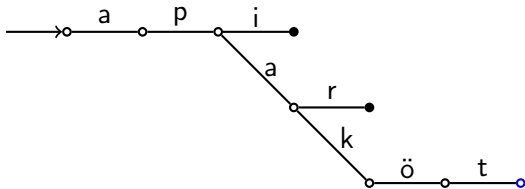




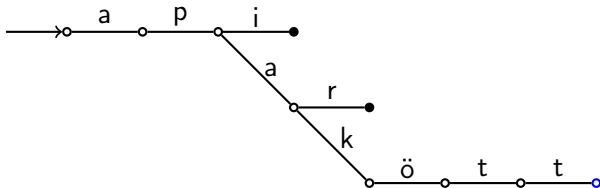
„ttur“



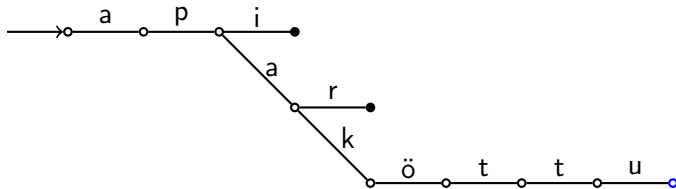
„tur“



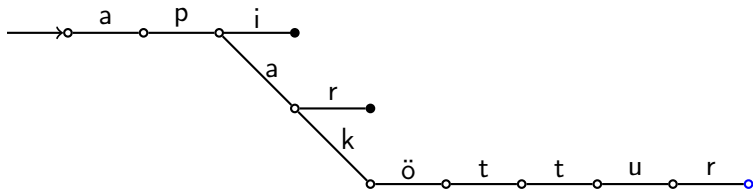
„ur“

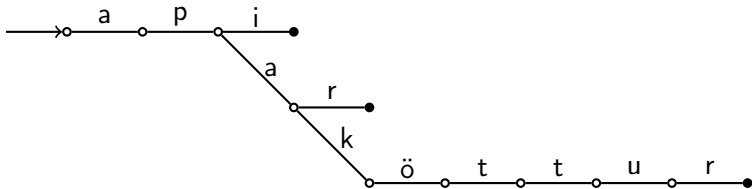


„r“

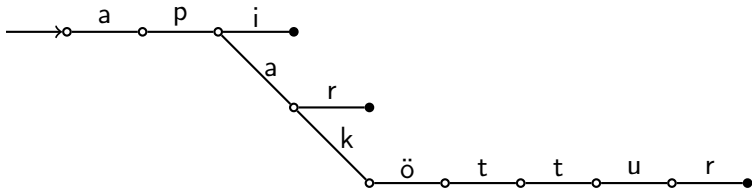


”  
”

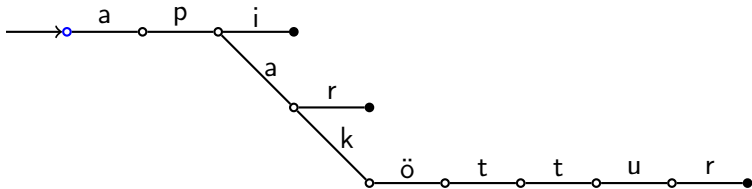




„altari“

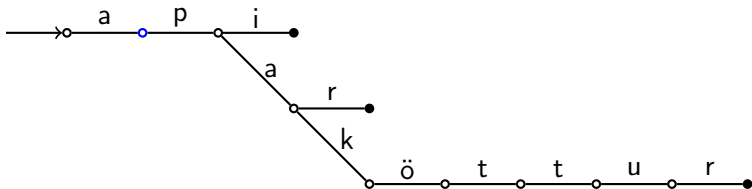


„altari“

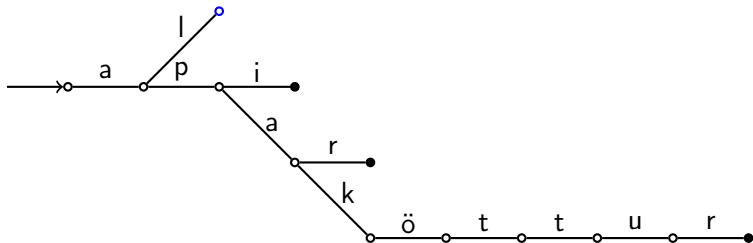




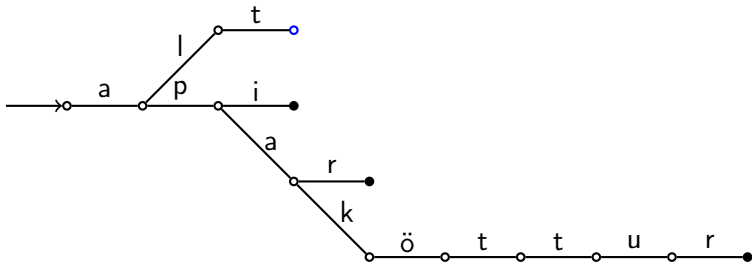
„ltari“



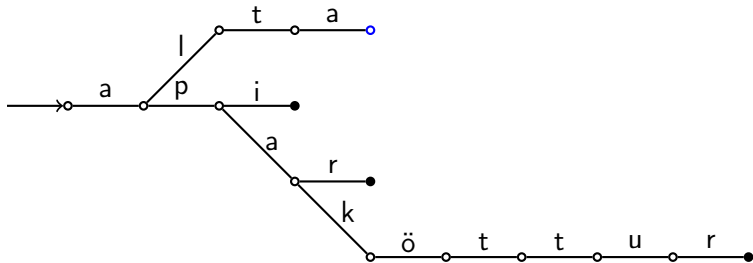
„tari“



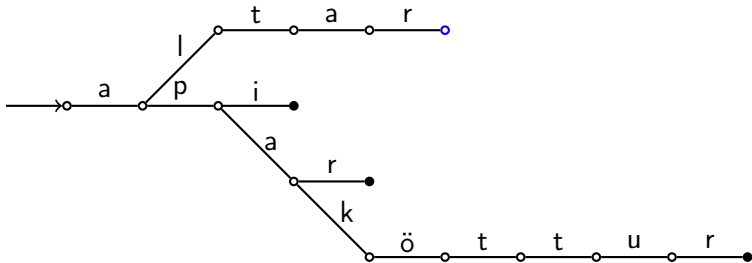
„ari“



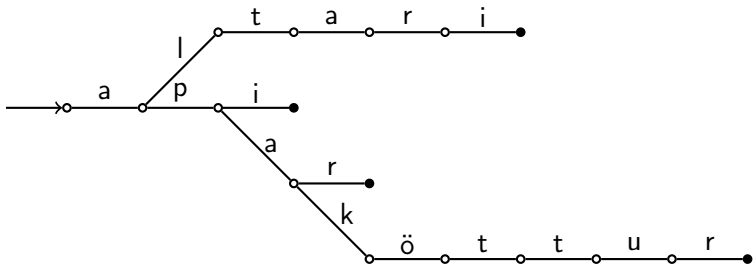
„ri“



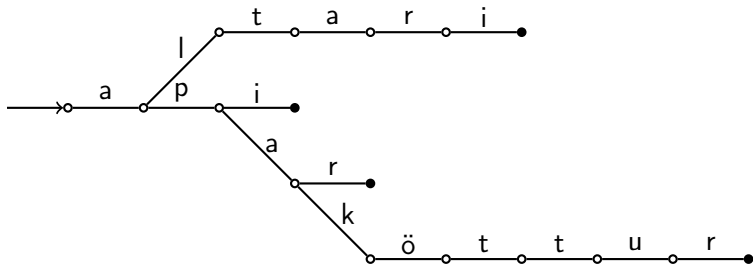
„i“





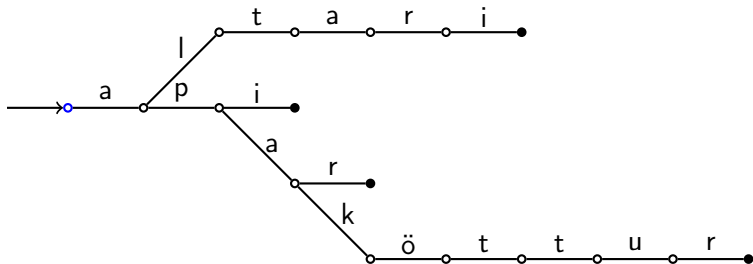


„apaspil“

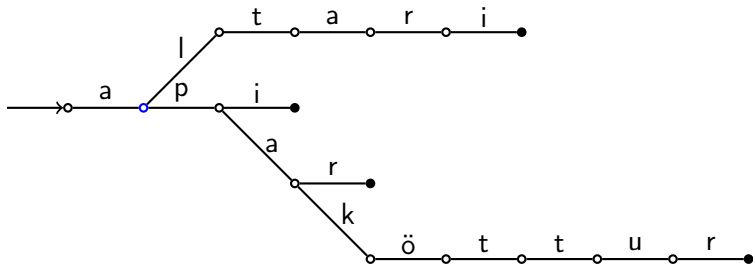




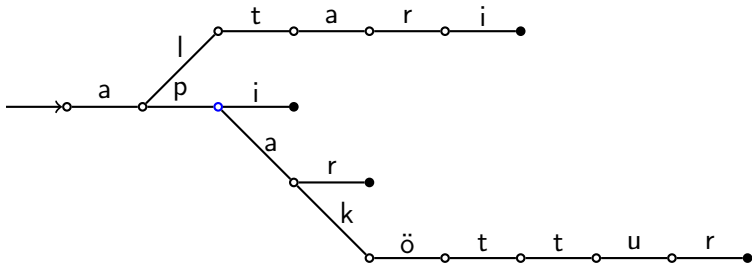
„apaspil“



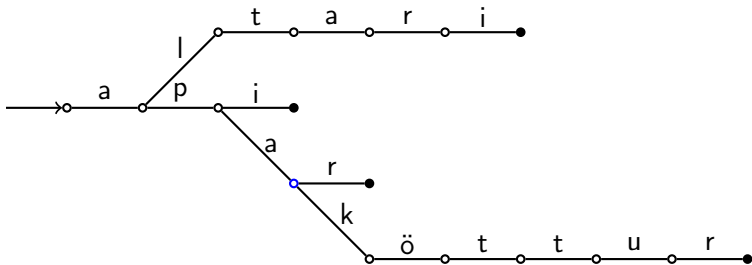
„paspil“



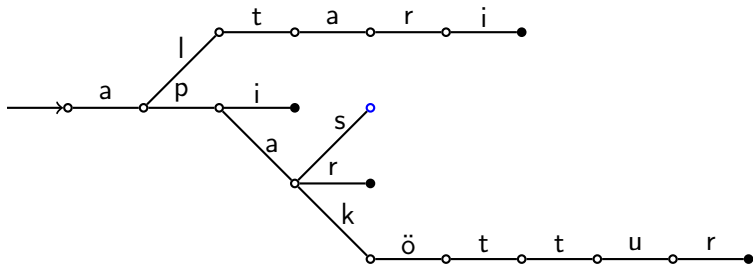
„aspil“



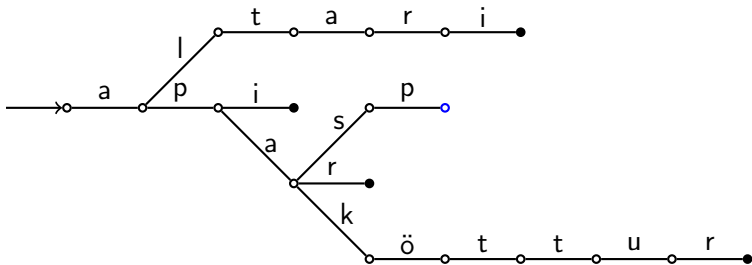
„spil“



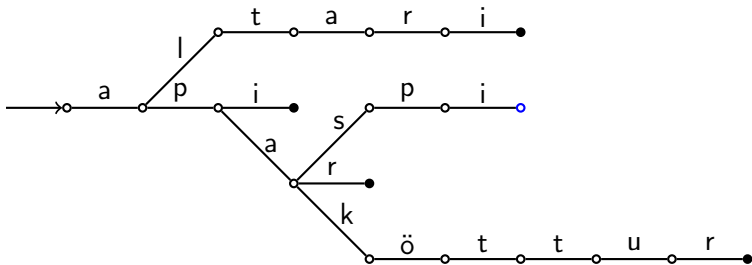
„pil“



„il“

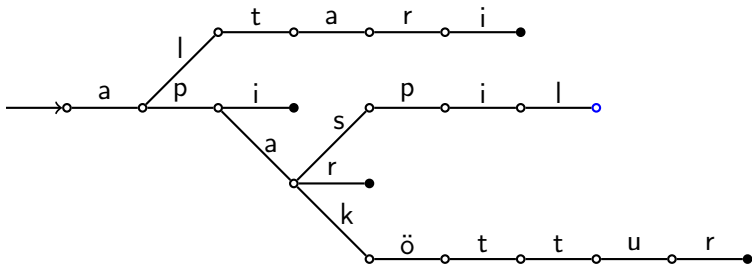


„I“

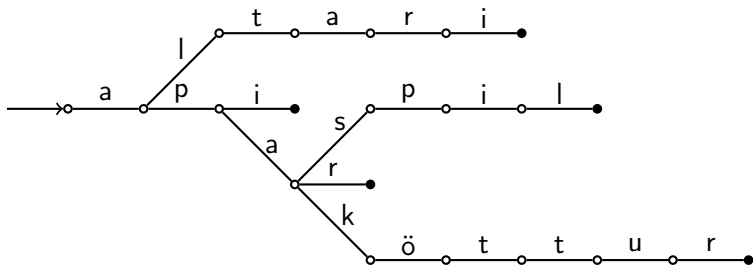


”

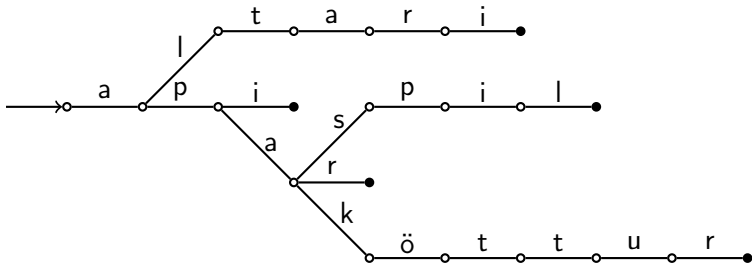
”



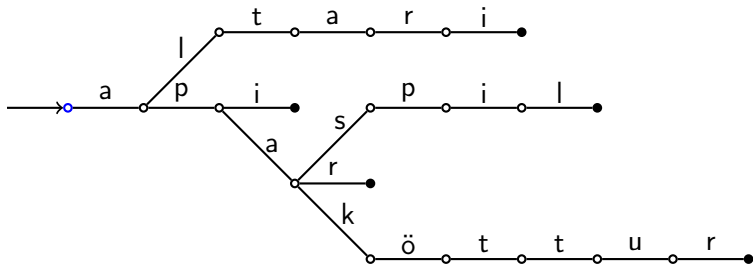




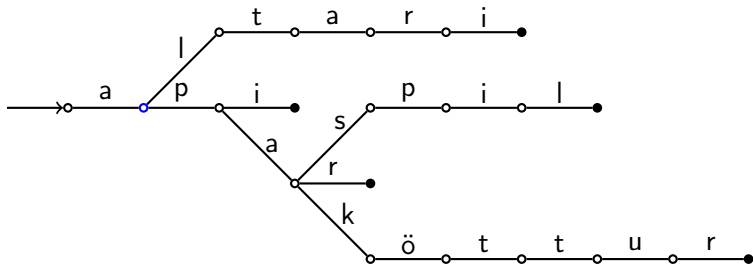
„altarisganga”



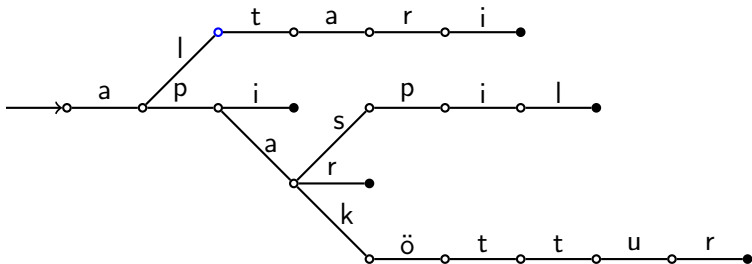
„altarisganga“



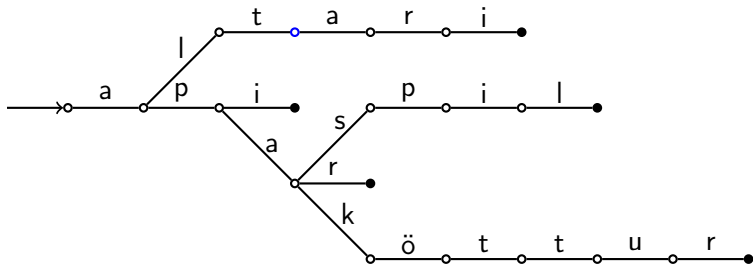
„ltarisganga“



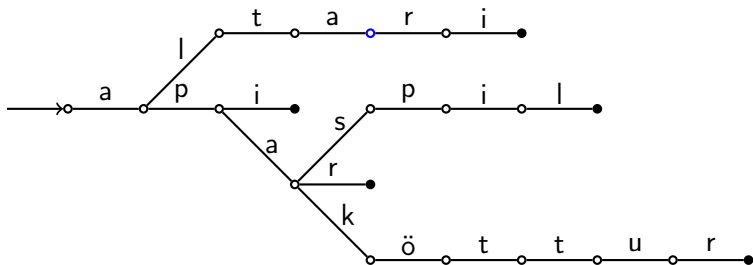
„tarisganga“



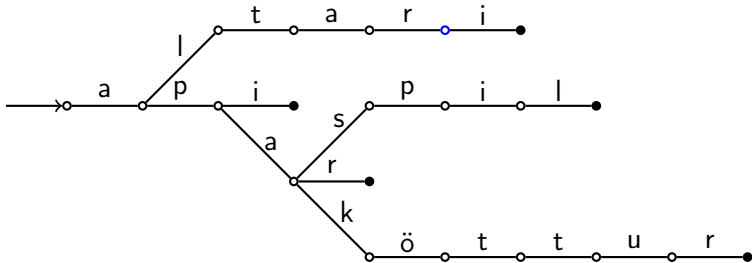
„arisganga“



„risganga“

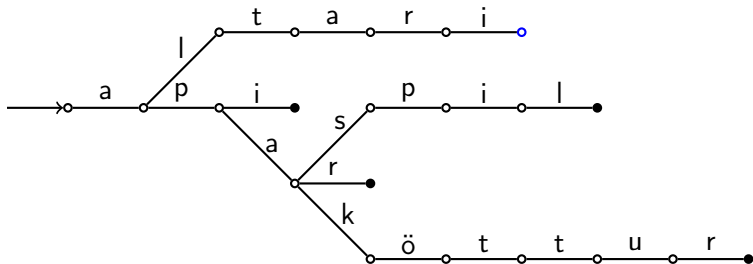


„isganga“

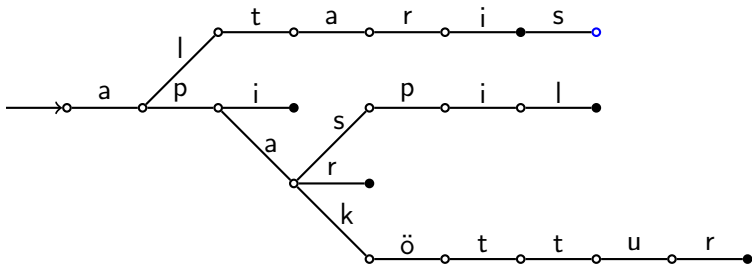




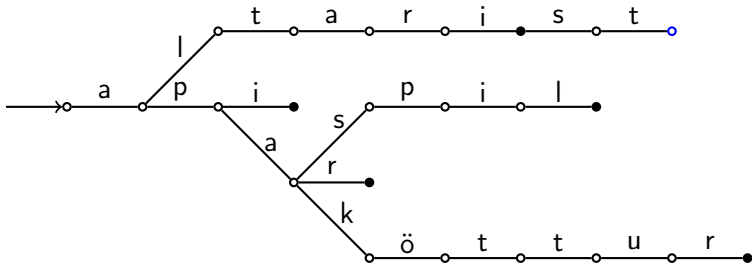
„sganga”



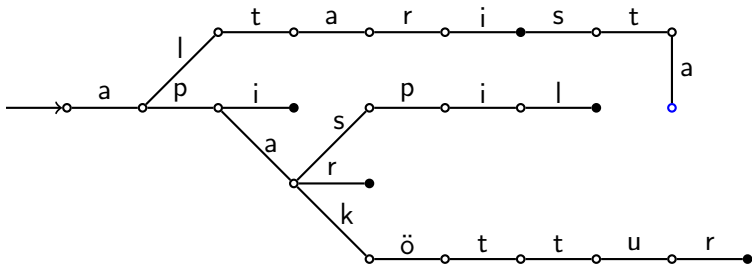
„ganga”



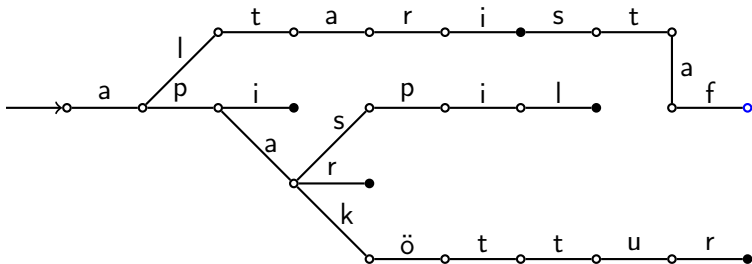
„anga”



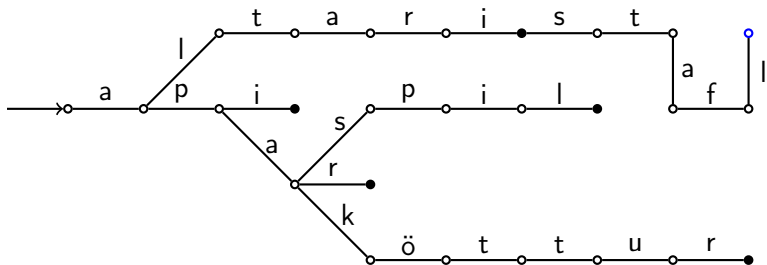
„nga”



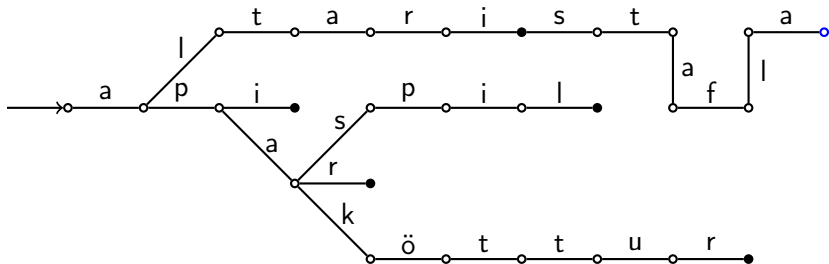
„ga“

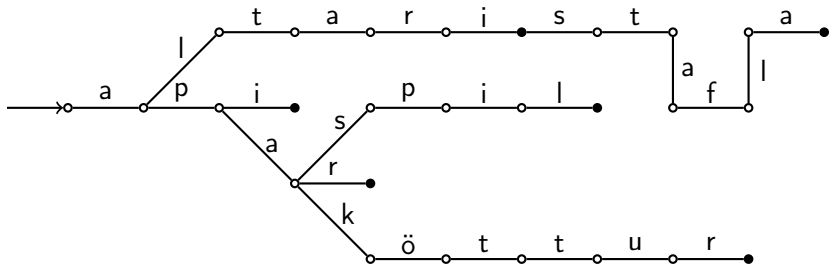


„a“



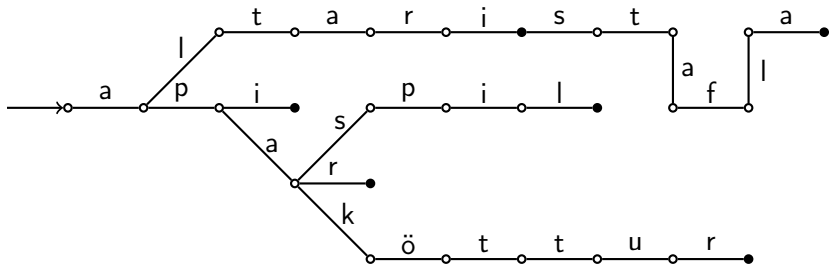
”  
”



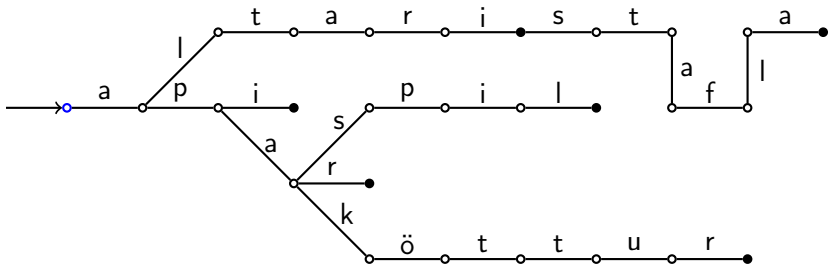




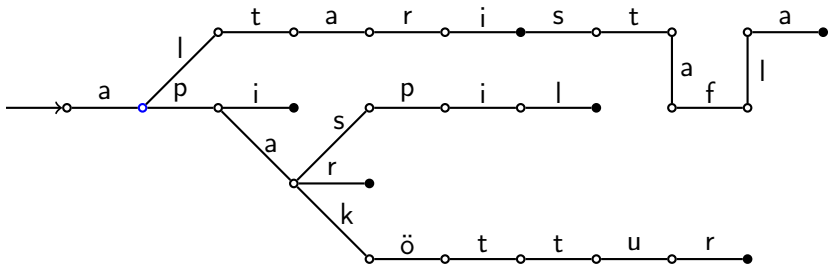
# „altarisganga“



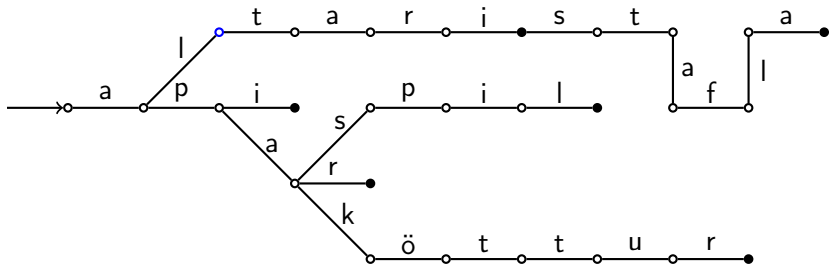
„altarisganga“



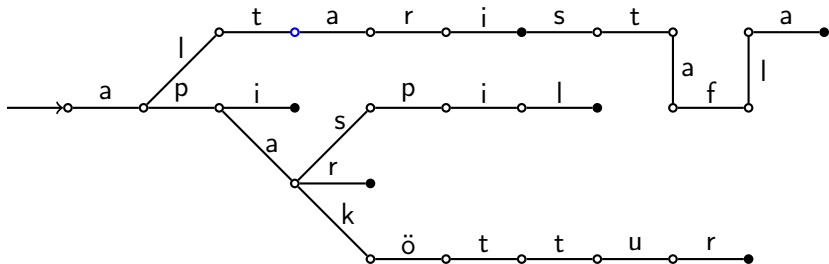
„ltarisganga“



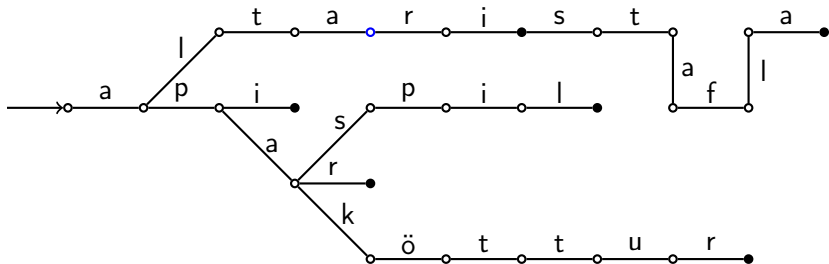
„tarisganga”



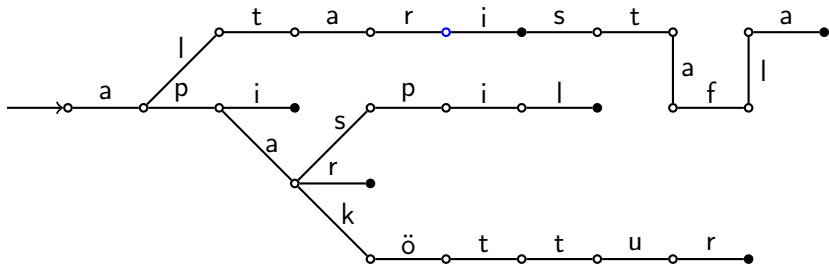
„arisganga”



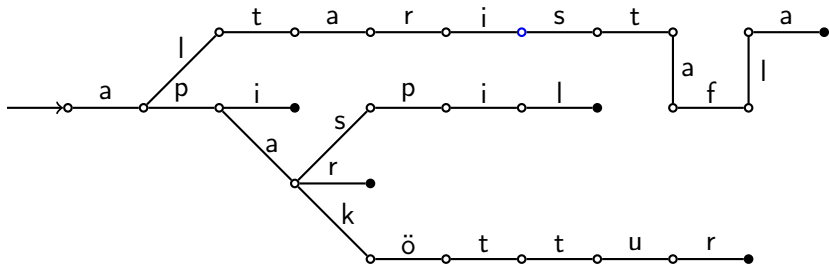
„risganga“



„isganga“

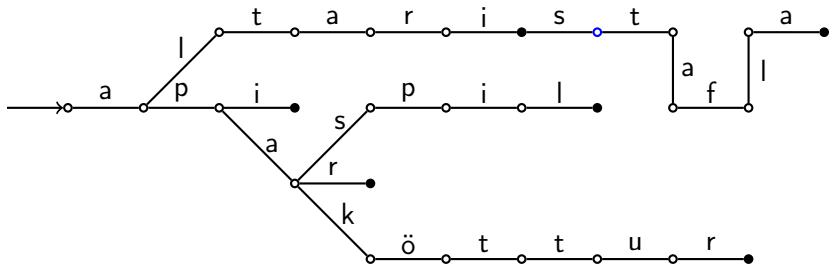


„sganga“

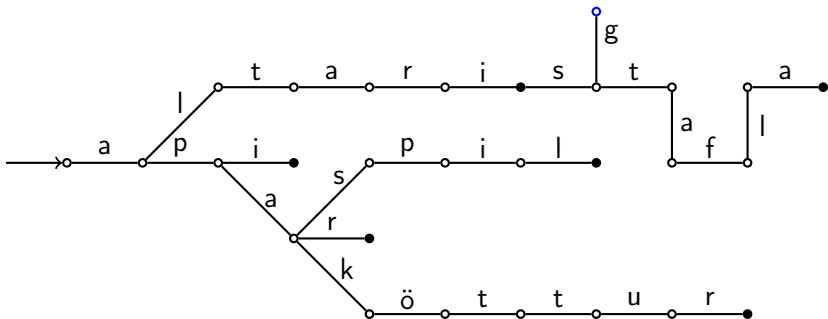




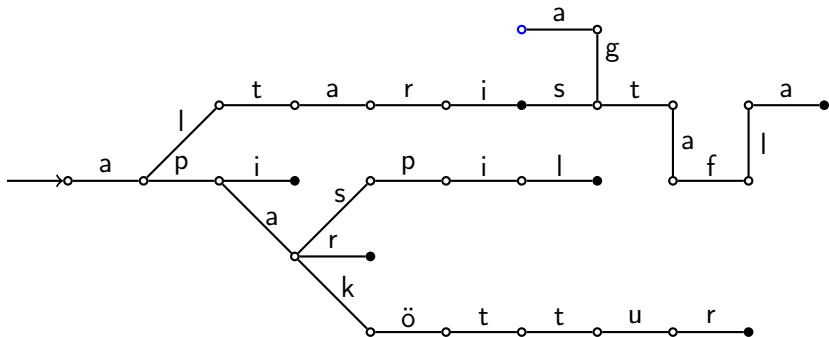
„ganga“

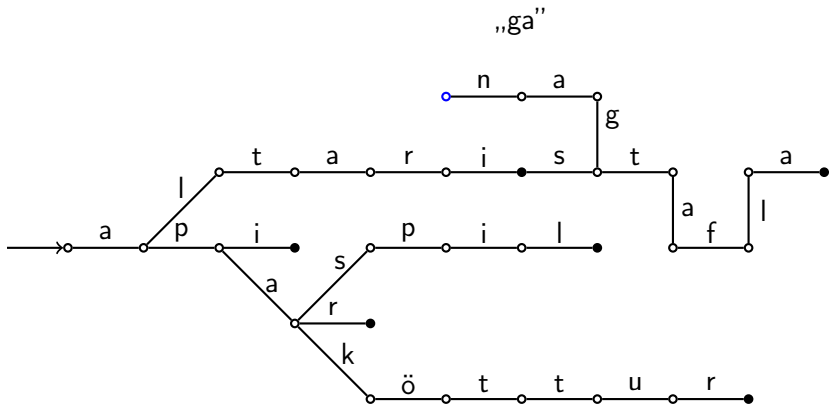


„anga”



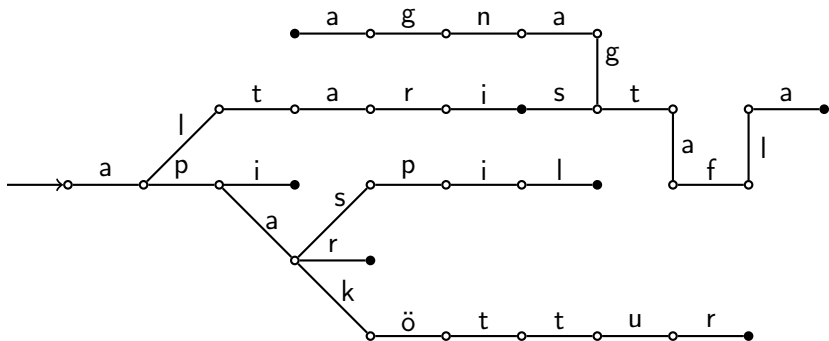
„nga“











- ▶ Til að útfæra forstrengstré munum við taka frá fylki af hnútum og úthluta þeim eftir þörf.



- ▶ Til að útfæra forstrengstré munum við taka frá fylki af hnútum og úthluta þeim eftir þörf.
- ▶ Með þessa aðferð í huga er gott að vita hvað munum þurfa marga hnúta í heildina.

- ▶ Til að útfæra forstrengstré munum við taka frá fylki af hnútum og úthluta þeim eftir þörf.
- ▶ Með þessa aðferð í huga er gott að vita hvað munum þurfa marga hnúta í heildina.
- ▶ Fyrir tómt forstrengstré þurfum við einn hnút.

- ▶ Til að útfæra forstrengstré munum við taka frá fylki af hnútum og úthluta þeim eftir þörf.
- ▶ Með þessa aðferð í huga er gott að vita hvað munum þurfa marga hnúta í heildina.
- ▶ Fyrir tómt forstrengstré þurfum við einn hnút.
- ▶ Ef við viljum bæta við streng þá þurfum við aldrei fleiri hnúta en lengd strengsins.

- ▶ Til að útfæra forstrengstré munum við taka frá fylki af hnútum og úthluta þeim eftir þörf.
- ▶ Með þessa aðferð í huga er gott að vita hvað munum þurfa marga hnúta í heildina.
- ▶ Fyrir tómt forstrengstré þurfum við einn hnút.
- ▶ Ef við viljum bæta við streng þá þurfum við aldrei fleiri hnúta en lengd strengsins.
- ▶ Svo heildarfjöldi hnúta þarf að vera einum meira en samtals lengd allra strengjanna sem við setjum í forstrengstréð.

```

6 #define ALPHABET 26
7 #define MAXN 1000000
8 typedef struct { int t[ALPHABET], v; } trienode;
9 typedef struct { int s, r; trienode m[MAXN]; listnode w[MAXN]; } trie;
10 int val(char c) { return c - 'a'; }
11 int trie_node(trie *t, int v)
12 {
13     int i;
14     for (i = 0; i < ALPHABET; i++) t->m[t->s].t[i] = -1;
15     t->m[t->s].v = -1;
16     return t->s++;
17 }
18 void trie_init(trie *t) { t->s = 0, t->r = trie_node(t, -1, -1); }
19
20 void trie_insert(trie *t, char *s, int x)
21 {
22     int h;
23     for (h = t->r; *s; h = t->m[h].t[val(*s++)])
24         if (t->m[h].t[val(*s)] == -1)
25             t->m[h].t[val(*s)] = trie_node(t, h, val(*s));
26     t->m[h].v = 1;
27 }

```

- ▶ Ef við viljum setja streng af lengd  $n$  inn í forstrengstréð þá tekur það  $\mathcal{O}(n)$ .

- ▶ Ef við viljum setja streng af lengd  $n$  inn í forstrengstréð þá tekur það  $\mathcal{O}(n)$ .

