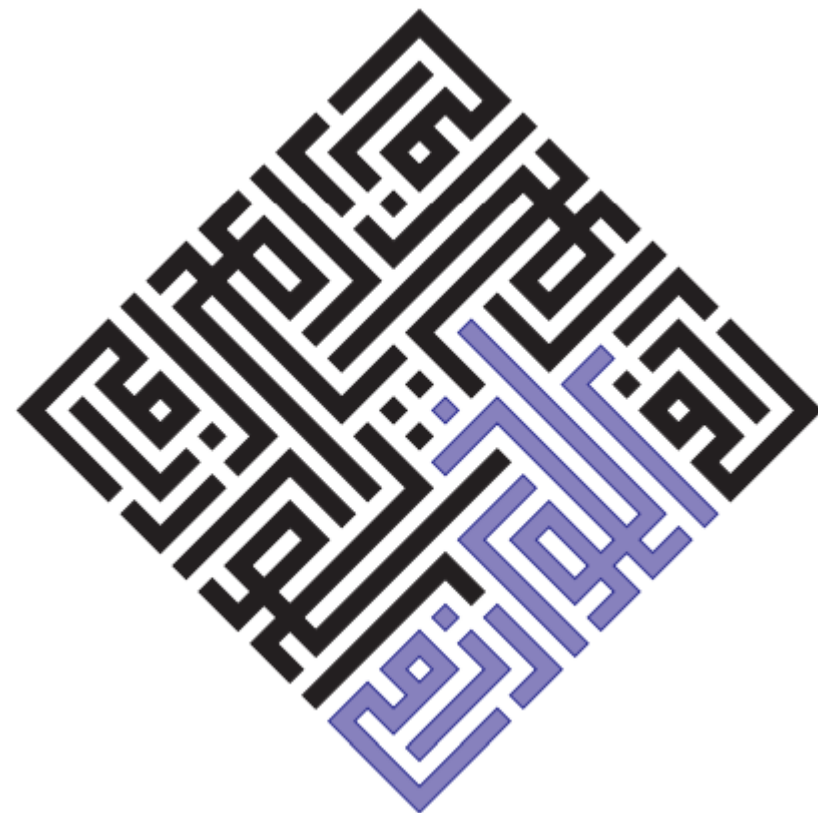


TÖL403G GREINING REIKNIRITA

14. Slembin reiknirit 2

Hjálmtyr Hafsteinsson
Vor 2022



- Gagnagrindin *treap*
 - Virkni
 - Snúningur í tvíleitartrjám
 - Aðgerðir á *treap*
 - Tímaflækja

DC 3.1

- Hafið séð margar leiðir til að geyma gögn:

- Fylki
 - Tengur listi
- } Einfaldar og ódýrar, en leit er hægðvirk

- Tvíleitartré (*BST*)

- AVL tré
 - Rauð-svart tré
 - Splay tré
 - B-tré
- } Útgáfur af tvíleitartrejám sem viðhalda jafnvægi

↑
Töluverður aukakostnaður við það að viðhalda jafnvæginu

Tvíleitartré eru einföld og góð gagnagrind "að meðaltali"

↑
Ef inntakið er slembið

Hugmynd:

Nota tvíleitartré, en færa slembnina inn í gagnagrindina!

Þá skiptir dreifing inntaksins ekki máli

- Blandar saman tvíleitartré og hrúgu (*heap*)

- Nafnið kemur frá *TREe* og *hEAP*

- Sett fram árið 1989 af Raimund Seidel og Cecilia Aragon (upphafleg grein)

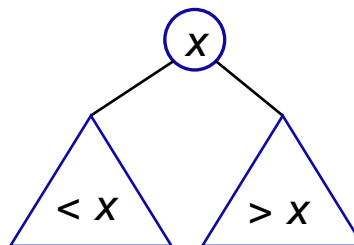
Trúga (TRé og hRÚGA)?

Doktorsleiðbeinandi Jeff Erickson, samnemandi minn í Cornell

Var doktorsnemandi Seidels, nú prófessor við Washington háskóla og flugakróbat(!)

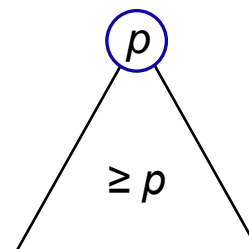
- Geymir lykla (*keys*)

- sem uppfylla tvíleitarskilyrði



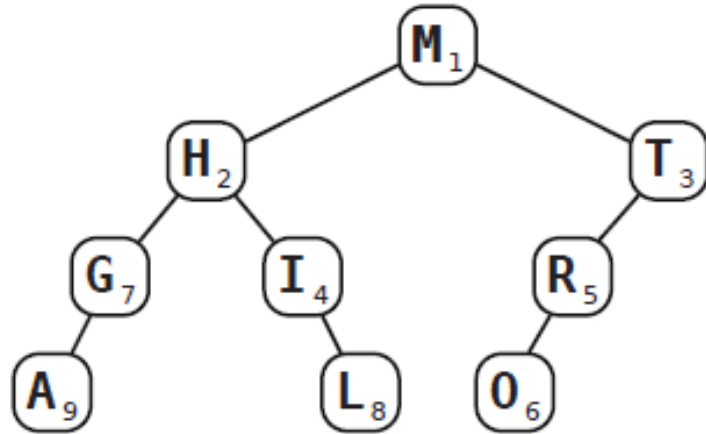
- Hver hnútur hefur forgang (*priority*)

- sem uppfylla hrúguskilyrði (*min-heap*)



Forgangur hnúta ákveðinn til með slemitölugjafa þegar hnúturinn er búinn til
Heiltala á bilinu 0 til $2^{32}-1$

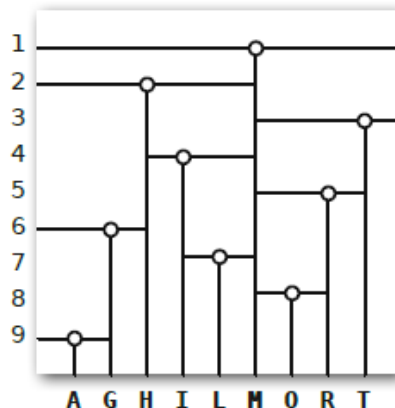
Dæmi um *treap*



Leyfum ekki endurtekna lykla
en forgangsgildi mega vera endurtekinn

Hægt að sýna að fyrir gefið mengi af lyklum
og gefið mengi af forgangsgildum þeirra þá
er aðeins til nákvæmlega eitt *treap*

Bókstafir eru lykjar, tölur eru forgangur



Önnur framsetning á sama *treap*:

Hnútarnir eru punktar í sléttu

Lyklarnir eru í röð á lárétta ásnum

Forgangsgildin eru á lóðrétta ásnum

Þessi framsetning hefur ýmsa
rúmfræðilega eiginleika, sem
ekki verður farið nánar út í

■ Innsetning á lykli x í *treap* T :

Búa til slembiforgang p_x

Leita að x í trénu T

Ef x finnst ekki þá

Setja hnút þess sem lauf í T

Snúa upp þar til p_x uppfyllir hrúguskilyrði

Slembiheiltala á bilinu 1 til $2^{32}-1$

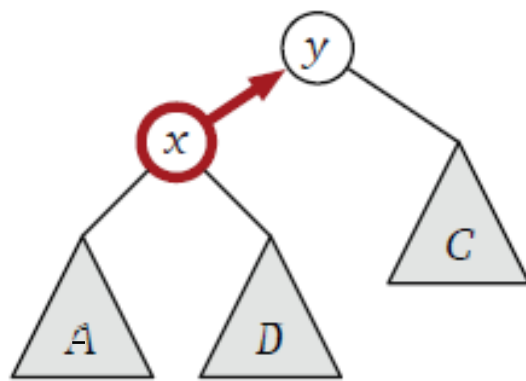
Eins og venjulega í tvíleitartrám

Nýir hnútar koma alltaf inn
sem lauf í tvíleitartrám

Snúningur (*rotation*) breytir lögun trésins
án þess að brjóta tvíleitaraskilyrði

Snúningur (*rotation*) í tvíleitartré

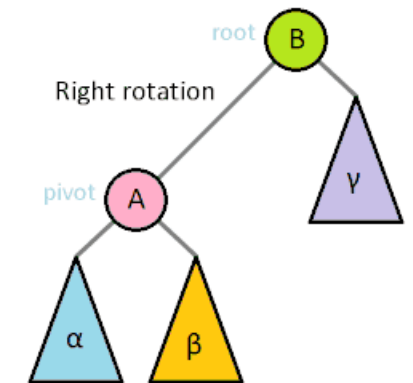
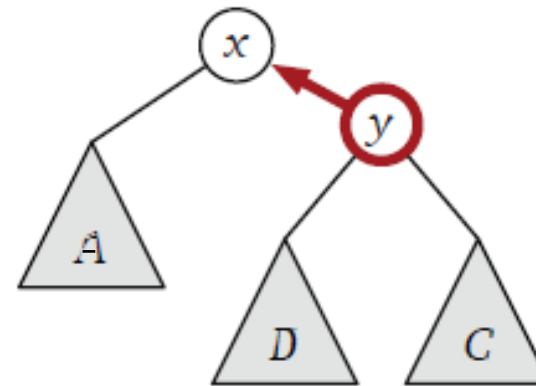
- Einföld aðgerð á tvíleitartré sem breytir lögun þeirra án þess að skemma tvíleitaraskilyrðið



Hægri snú



Vinstri snú



Hreyfimynd frá Wikipedia

Hér gildir:

$$A < x < B < y < C$$

Hér gildir:

$$A < x < B < y < C$$

Framkvæmd:

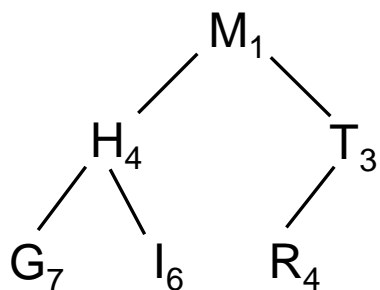
Láta D verða vinstra barn y
Láta y verða hæggra barn x
Láta foreldri y benda á x í stað y

Framkvæmd:

Láta D verða hæggra barn x
Láta x verða vinstra barn y
Láta foreldri x benda á y í stað x

Hver snúningur er
 $O(1)$ tíma aðgerð

Dæmi um innsetningu



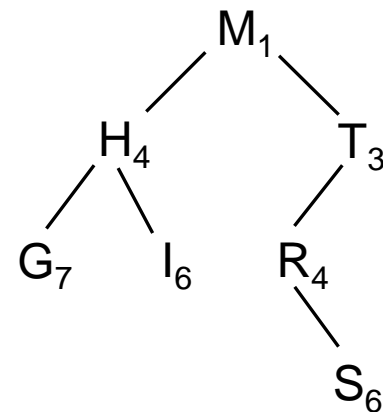
Upphaflegt tré

Setja lykilinn S inn í þetta *treap*

Köllum á slembitölugjafa:

Segjum að við fáum forganginn 6

Rekjum okkur niður tréð
og setjum S inn sem lauf:



Forgangurinn er í lagi, því $6 \geq 4$, svo við erum búin

Dæmi um innsetningu, frh.

Hvað ef slembitölugjafinn hefði skilað 2?

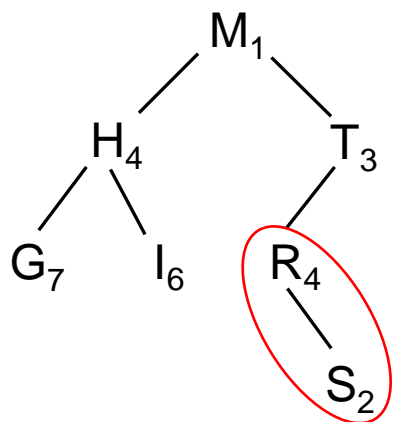
Þá brýtur þessi nýi hnútur gegn hrúguskilyrðinu

Þurfum að snúa honum upp þar til hann gerir það ekki lengur

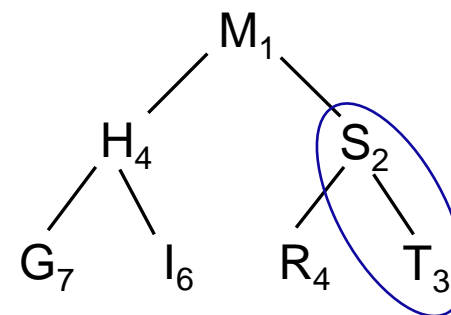
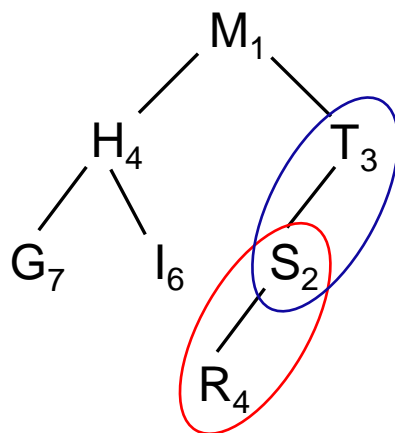
Nýi hnúturinn hækkar um eitt lag við hvern snúning

Gætum þurft að fara upp í rót

Gerum vinstri snúning á S:



Gerum hægri snúning á S:



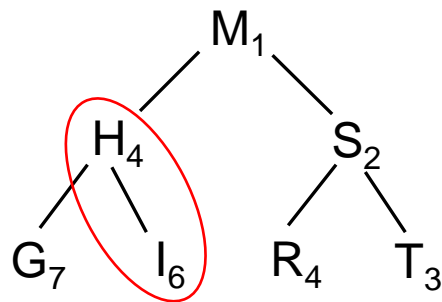
■ Framkvæmum innsetningu "afturábak"

- Ef hnúturinn er ekki lauf, þá snúum við honum niður þannig að hann verði lauf
- Klippum svo hnútinna út

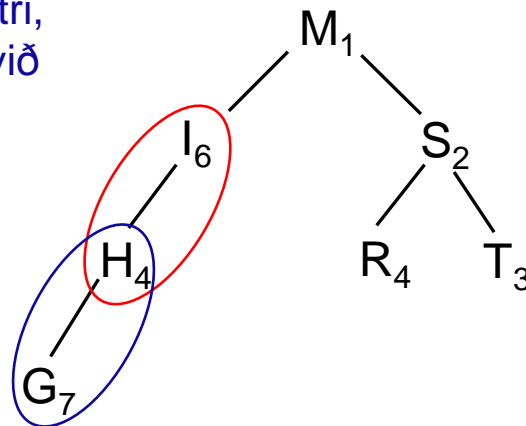
Notum vinstri/hægri snúning eftir því hvort barnið hefur lægra forgangsgildi

■ Dæmi:

Eyða hnútinum H úr þessu tré:

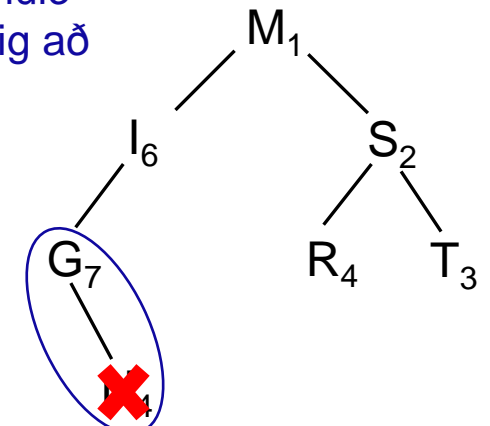


Snúum H til vinstri, annars brjótum við hrúguskilyrðið



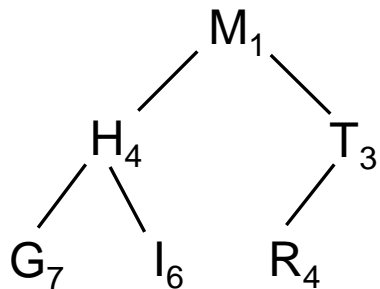
Hér er H aðeins með eitt barn og þá gætum við tengt framhjá honum

Eða við getum snúið H til hægri, þannig að það verði lauf



Hendum svo hnútinum

- Setja lykilinn V með forgangsgildinu 0 (V_0) inn í þetta *treap*:



Skipting (*split*)

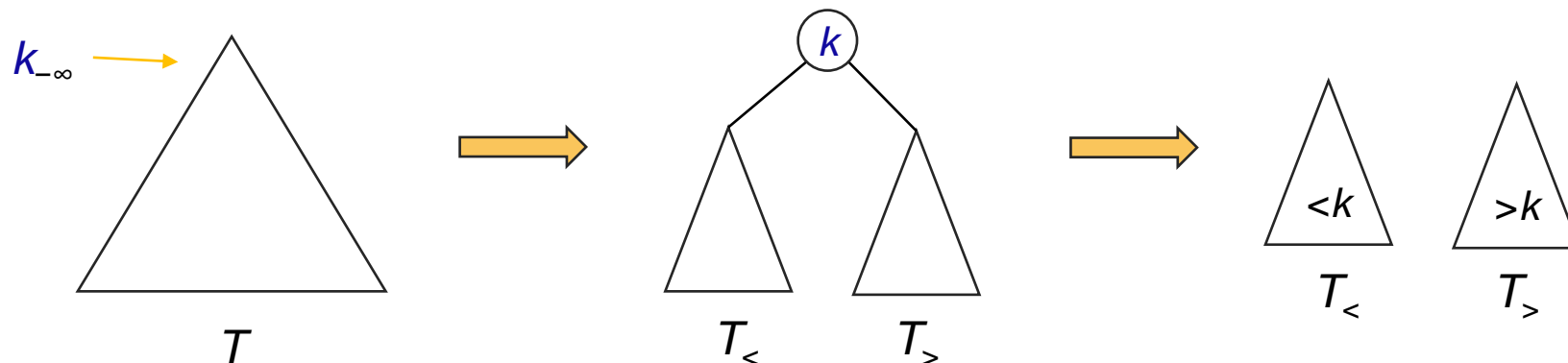
- Aðgerðin skiptir *treap* T upp í $T_{<}$ og $T_{>}$
 - Allir lykjar í $T_{<}$ hafa lægri lykilgildi en eitthvað gefið gildi k
 - Allir lykjar í $T_{>}$ hafa hærra lykilgildi en k

Reiknirit

Setja inn í T hnút með lykil k og forgang $-\infty$

Þessi nýi hnútur verður að rót trésins

Hendum hnútinum í burtu, þá eru vinstra hluttré $T_{<}$ og hæggra hluttré $T_{>}$

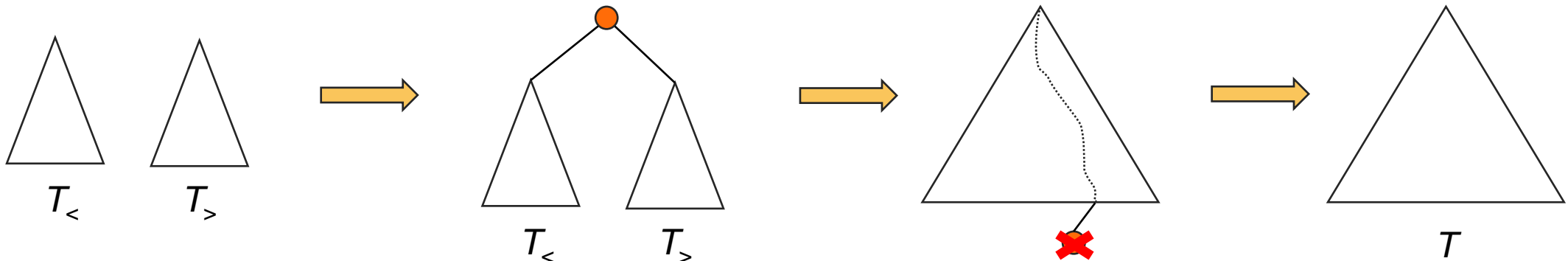


Sameining (*join*)

- Höfum tvö *treap* $T_<$ og $T_>$
 - Allir lykjar í $T_<$ eru minni en allir lykjar í $T_>$
- Sameinum þau í eitt *treap*

Reiknirit

Búa til gervihnút og setja $T_<$ sem vinstra barn hans og $T_>$ sem hægra barn
Snúa hnútinum niður þar til hann er lauf
Hendum hnútinum í burtu



- Allar aðgerðirnar:
 - Leitun (*search*) ← Eins og venjuleg leit að lykli í tvíleitartré
 - Innsetning (*insertion*)
 - Eyðing (*deletion*)
 - Skipting (*split*)
 - Sameining (*join*)
- taka tíma $O(dýpi(v))$, þar sem v er hnúturinn með lyklinum k
 - Ef lykillinn k er ekki í trénu þá er þetta $O(dýpi(v^-))$ eða $O(dýpi(v^+))$
 - Hnúturinn v^- inniheldur næsta gildi fyrir neðan k og v^+ inniheldur næsta gildi fyrir ofan k
- Í versta tilfelli er $dýpi(v) = O(n)$ ← Ef tréð er keðja af n hnútum og við erum að vinna með neðsta hnútinn

- Hægt að sýna (3.1.3 í [bók](#)) að vænt dýpi sérhvers hnútar er $O(\log(n))$
- Þýðir að væntur tími allra aðgerða á *treap* er $O(\log(n))$

Treap hegðar sér eins og tvíleitartre þar sem innsetningaröðin er slembin

Oftast er inntaksröðin ekki slembin

en *treap* nær þeim áhrifum með því að færa slembnina inn í gagnagrindina

Höfum því væntan tíma aðgerða $O(\log(n))$, alveg sama hver dreifing inntaksins er

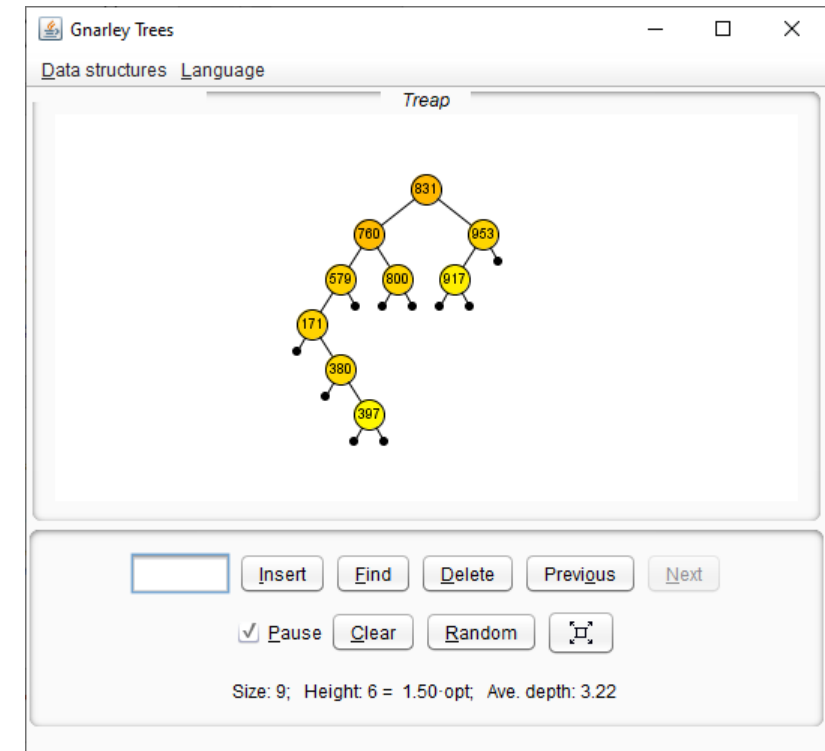
Með því að búa til slembið forgangsgildi og nota það til að stýra skipulagi trésins

Ekki hræðast slembnina!
Nýtum okkur eiginleika hennar

Sérstaklega mikilvægt fyrir streymisreiknirit (*on-line algorithms*) þar sem inntakið birtist smátt og smátt

Hermun á treap á Gnarley trees

- Gnarley trees er vefsíða sem sýnir virkni margra gagnagrinda
- Hægt að rekja sig í gegnum framkvæmd einstakra aðgerða
- Í *treap* hermun er forgangurinn sýndur sem litur
 - Gulur litur er hátt forgangsgildi, en appelsínugult er lágt gildi
 - Ekki hægt að ráða forgangsgildinu, það er slembið
- Lykilgildi eru heiltölur frá 1 til 999
 - Hægt að setja inn lykla á því bili (ef ekki fyrir í trénu)
- Hægt að búa til *treap* af tiltekinni stærð
 - Setja fjölda hnúta í inntakshólfið og smella á hnappinn *Random*



1. Setjið lyklana A (8), B (2) og C (6) inn í *treap*. Tölurnar í svigunum eru forgangur (*priority*) lyklanna.
2. Við setjum n lykla inn í *treap* í hækkandi röð. Hvernig þyrfti forgangur einstakra lykla að vera til þess að tréð yrði hægri keðja (þ.e. allir hnútar bara með hægri barn).
3. Í *treap* hefur rótin lægsta forgangsgildið. Hvar getur þriðja-minnsta forgangsgildið verið?