TÖL403G GREINING REIKNIRITA

11. Gráðug reiknirit 1

Hjálmtýr Hafsteinsson

Vor 2022

Í þessum fyrirlestri

- Að raða skrám á segulband (*storing files on tape*)
  - Jafnar líkur á skrám
  - Mismunandi lestrartíðni

- Skipulag námskeiða (verkval, *scheduling classes*)
  - Finna mesta fjölda námskeiða sem skarast ekki

4.1 − 4.3

- Almennt form sannana á gráðugum reikniritum

Skrár á segulbandi

- Höfum *n* skrár (*files*) sem á að geyma á segulbandi
- Skrá *i* hefur lengd *L*[*i*]
  - Þurfum að spóla framhjá skrám 1 til *k*−1 til þess að lesa skrá *k*

$$\text{kostn}(k) = \sum_{i=1}^{k} L[i]$$

þarf að spóla framhjá
1 til *k*−1 og lesa skrá *k*

- Ef jafnlíklegt er að allar *n* skrárnar séu lesnar þá er <u>væntur kostnaður</u> (*expected cost*) að lesa skrá:

$$E[kostn] = \sum_{k=1}^{n} \frac{kostn(k)}{n} = \sum_{k=1}^{n} \sum_{i=1}^{k} \frac{L[i]}{n}$$

Umröðun skráa

- Getum breytt væntum kostnaði með því að umraða skránum
  - Lát π vera umröðun (*permutation*) á 1, ..., *n*. Þá gildir:

$$E[kostn(\pi)] = \sum_{k=1}^{n} \sum_{i=1}^{k} \frac{L[\pi(i)]}{n}$$

Hvaða umröðun á að nota?

- Dæmi:

A *B* *C*

L = [5, 1, 3]

A	*B*	*C*

Væntur kostnaður: $\frac{1}{3}(5+6+9) = \frac{20}{3}$

Betri röð:

B	*C*	*A*

Væntur kostnaður: $\frac{1}{3}(1+4+9) = \frac{14}{3}$

Ekki gott að hafa lengstu skrárnar fremst, þurfum alltaf að fara framhjá þeim til að komast í hinar

Besta röðunin

- Setning: $E[kostn(\pi)]$ er í lágmarki þegar

$$L[\pi(i)] \leq L[\pi(i+1)] \quad \text{fyrir } i = 1, \ldots n-1$$

skrárnar eru í hækkandi röð eftir lengd

- Sönnun: (með mótsögn)

Segjum að umröðunin π gefi lágmarks væntan kostnað, en samt gildi að $L[\pi(i)] > L[\pi(i+1)]$ fyrir eitthvert i.

	a	b	
	$\pi(i)$	$\pi(i+1)$	

Hvað myndi gerast ef við víxluðum á skrám *a* og *b* á segulbandinu?

Kostnaðurinn við að lesa *a* myndi aukast um $L[b]$

Því nú er *b* fyrir framan *a* og þurfum að fara framhjá henni fyrst

og kostnaðurinn við að lesa *b* myndi minnka um $L[a]$

Því nú er *a* ekki fyrir framan *b* og sleppum við að fara framhjá henni

Sönnun, frh.

Heildarbreytingin á væntum kostnaði væri $\frac{L[b] - L[a]}{n}$

En við gáfum okkur að *L*[*a*] > *L*[*b*], svo að þetta er <u>lækkun</u> á væntum kostnaði

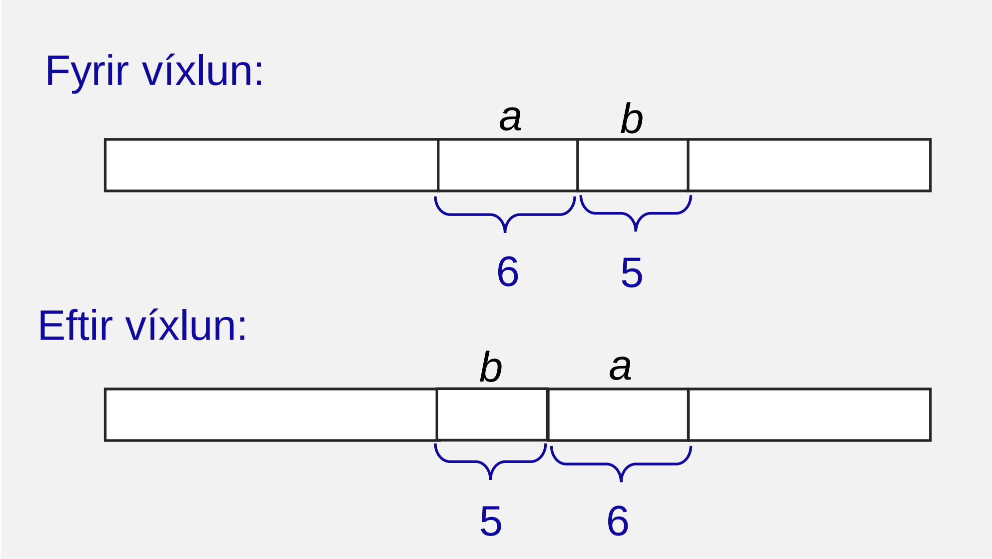
Hvernig er hægt að lækka eitthvað sem er þegar í lágmarki?

En kostnaðurinn var þegar í lágmarki, svo þetta er í mótsögn við það!

Forsendan sem við gáfum okkur hlýtur því að vera röng

þ.e. að til sé i þannig að $L[\pi(i)] > L[\pi(i+1)]$

Þetta þýðir að ef væntur kostnaður er í lágmarki
þá hljóta skrárnar að vera í hækkandi lengdarröð



Gráðugt reiknirit er best!

- Hér er þá komið gráðugt reiknirit sem finnur alltaf bestu lausn á verkefni
- Óvenjulegt að gráðug reiknirit finni alltaf bestu lausn
  - Oftast finna þau bara góða lausn, eða staðvært bestu (*locally optimal*) lausn
- Græðgin getur verið á ýmsa vegu:
  - Hér felst hún í að setja stystu skránna fremst og leysa svo restina á sama hátt

- Tími: $O(n \log(n))$ til að raða skránum í hækkandi röð eftir lengd

Dæmigerð tímaflækja fyrir gráðug reiknirit.
Þau eru oftast nokkuð hraðvirk: $O(n)$ eða $O(n \log(n))$

Önnur tíðnidreifing

- Hvað ef skrárnar eru lesnar með mismunandi tíðni?
  - Gefið fylkið *F*[1..*n*] þar sem *F*[*i*] er lestrartíðni á skrá *i*

Þá myndum við vilja hafa þær skrár sem eru lesnar oft framarlega á segulbandinu

og það eru ekki endilega stystu skrárnar!

- Nú er heildarkostnaðurinn:

$$\Sigma kostn(\pi) = \sum_{k=1}^{n} \left(\underbrace{F[\pi(k)]}_{\text{tíðni lestrar á skrá } k} \cdot \underbrace{\sum_{i=1}^{k} L[\pi(i)]}_{\text{kostnaður við lestur á skrá } k} \right) = \sum_{k=1}^{n} \sum_{i=1}^{k} (F[\pi(k)] \cdot L[\pi(i)])$$

Besta röðun fyrir almenna tíðnidreifingu

- Sjáum:

  Best að hafa skrár með háa tíðni fremst - hátt $F[i]$ fremst

  Best að hafa skrár með stutta lengd fremst - lágt $L[i]$ fremst

  Skoðum hlutfallið L[i]/F[i]:

  Best að hafa skrár með lágt gildi á L[i]/F[i] fremst

- Setning:

  Heildarkostnaðurinn $\Sigma kostn(\pi)$ er í lágmarki þegar

$$\frac{L[\pi(i)]}{F[\pi(i)]} \leq \frac{L[\pi(i+1)]}{F[\pi(i+1)]} \qquad \text{fyrir } i = 1, \ldots, n-1$$

- Sönnun:

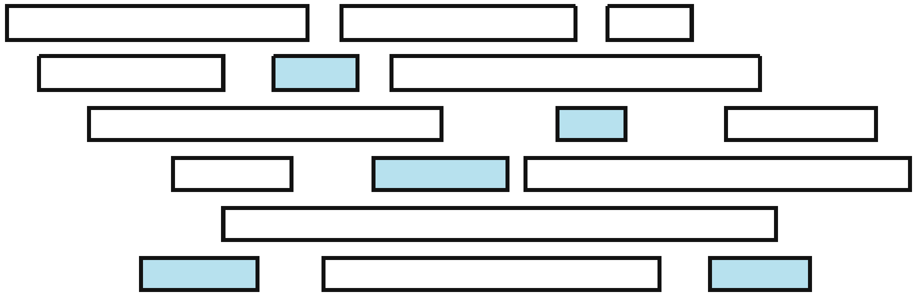
  Svipuð röksemdafærsla og í fyrri sönnun (með mótsögn)

Skipulag námskeiða (*scheduling classes*)

- Höfum n námskeið með upphafstíma $S[1..n]$ og lokatíma $F[1..n]$
  - Gildir alltaf að $S[i] < F[i]$

- Viljum velja sem stærsta hlutmengi $X \subseteq \{1, 2, ..., n\}$, þ.a. fyrir sérhver stök $i, j \in X$ gildir að $S[i] > F[j]$ eða $S[j] > F[i]$

i er á eftir *j*   *j* er á eftir *i*



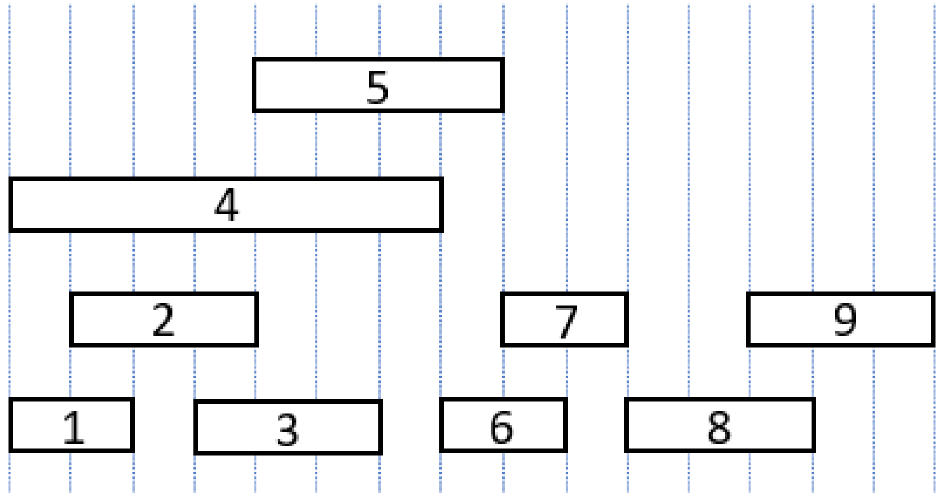
- Viljum finna mesta fjölda námskeiða sem ekki skarast
  - Hugsum ekkert um lengd námskeiðana, bara fjöldann

Gætum líka talað um verk (*activity*) í staðinn fyrir námskeið

Æfing

- Gefin þessi 9 verk, hver er mesti fjöldi verka sem ekki skarast?

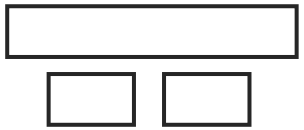
- Finnið tvær ólíkar lausnir



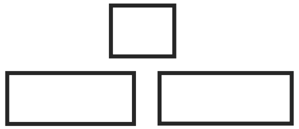
Möguleg gráðug reiknirit

- Hvernig á græðgin að vera?

- Velja fyrst lægsta upphafstíma *S*[*i*]? **Nei**, hægt að finna mótdæmi

Til dæmis: 

- Velja fyrst minnstu lengd námskeiðs (*F*[*i*] − *S*[*i*])? **Nei**, hægt að finna mótdæmi

Til dæmis: 

- Velja fyrst lægsta lokatíma *F*[*i*]? **Já**, þetta virkar 

Gráðugt reiknirit

Röðum verkum eftir lokatíma *F*[.]

Gerum nú ráð fyrir að verkin séu í röð eftir lokatíma, þ.a. verk 1 hefur lægsta lokatíma, verk 2 næstlægsta, o.s.frv.

Velja fyrsta verkið

Fyrir *i* = 2 **til** *n*

  Ef verk *i* skarast ekki við síðasta valið verk **þá**

    Velja verk *i*

Formlega:

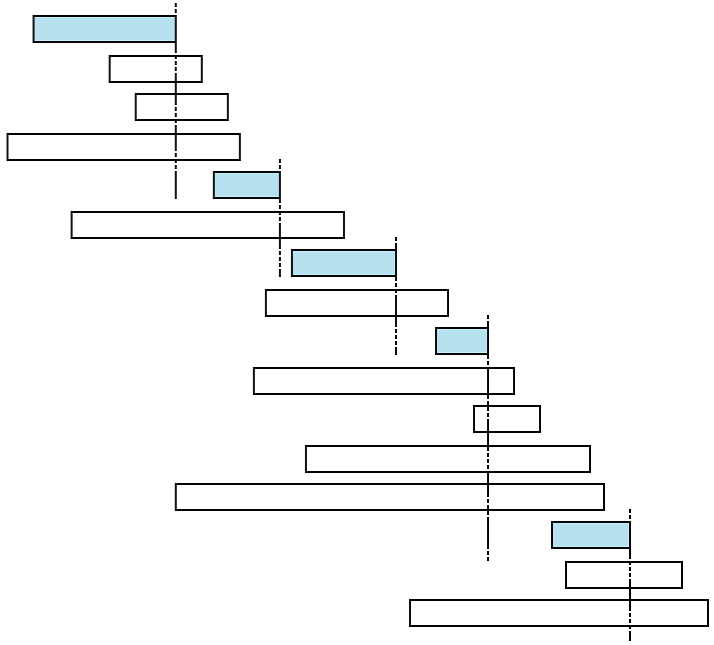
```
GreedySchedule(S[1..n], F[1..n]):
  sort F and permute S to match
  count ← 1
  X[count] ← 1
  for i ← 2 to n
      if S[i] > F[X[count]]
          count ← count + 1
          X[count] ← i
  return X[1..count]
```

Raða verkunum fyrst

Valin verk koma í fylkið *X*

Ef upphafstími verks *i* er hærri en lokatími síðasta valins verks ...

... þá getum við valið verk *i*



Bláu verkin valin

Reikniritið gefur bestu lausn

- Setning:

  Þetta gráðuga reiknirit gefur mesta fjölda námskeiða sem skarast ekki

- Sönnun:

  Segjum að gráðuga reikniritið skili eftirfarandi runu af verkum:

  $$G: \ \langle g_1, g_2, \ldots, g_k \rangle$$

  í röð eftir upphafstíma

  Segjum nú að til sé annað skipulag sem er betra

  Kannski byrjar það eins og *G*, en svo breytist það:

  $$O: \ \langle g_1, g_2, \ldots, g_{j-1}, c_j, c_{j+1}, \ldots, c_m \rangle \quad \text{með } m > k$$

  $c_j \neq g_j$ og er fyrsta verkið sem er öðruvísi

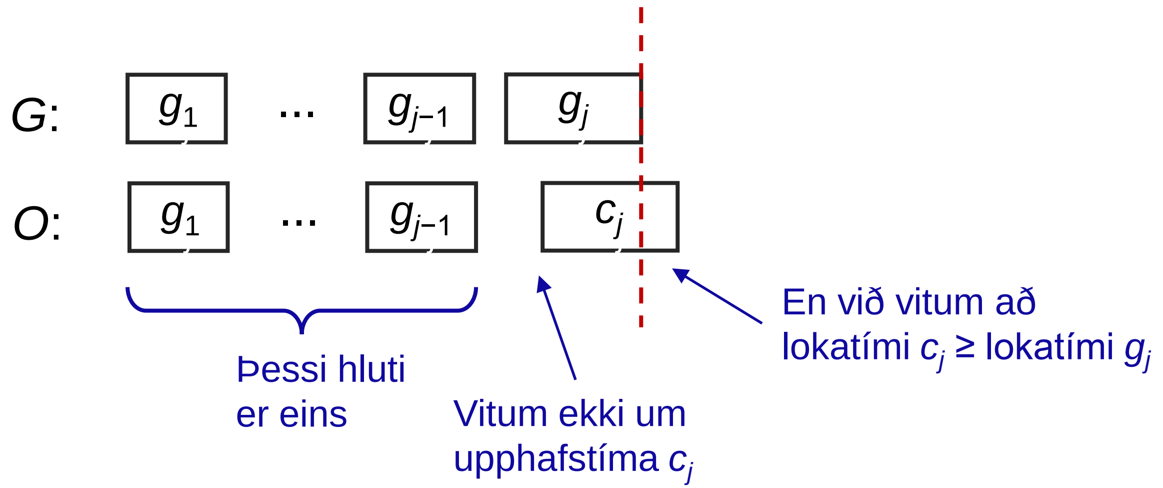
Sönnun, frh.

Gráðug: G: $<g_1, g_2, \ldots, g_{j-1}, g_j, g_{j+1}, \ldots, g_k>$

Besta: O: $<g_1, g_2, \ldots, g_{j-1}, c_j, c_{j+1}, \ldots, c_m>$

Munið að gráðuga reikniritið tók námskeiðin fyrir í hækkandi röð eftir lokatíma og g_j var valið því það var fyrsta verkið sem ekki skaraðist við g_{j-1}

En c_j skarast heldur ekki við g_{j-1} og $c_j \neq g_j$, svo þá hlýtur lokatími c_j að vera hærri en g_j



Við getum því sett g_j **í stað** c_j í verkrununa O, án þess að það skemmi neitt

O: $<g_1, g_2, \ldots, g_{j-1}, g_j, c_{j+1}, \ldots, c_m>$

Sönnun, frh.

Gráðug: G: $<g_1, g_2, ..., g_{j-1}, g_j, g_{j+1}, ..., g_k>$

Besta: O: $<g_1, g_2, ..., g_{j-1}, g_j, c_{j+1}, ..., c_m>$

Sáum að við gátum sett g_j í stað c_j í O

Getum þá haldið þessu áfram og sett g_{j+1} í stað c_{j+1} (ef þau eru ólík), o.s.frv.

Það er því til besta verkruna sem inniheldur **öll verk** sem gráðuga reikniritið valdi

En það þýðir að $k = m$, því ef það væri til verk c_{k+1} sem skarast ekki við neitt verk þá myndi gráðuga reikniritið velja það!

Við höfum þá sýnt að verkrunan G er alveg eins góð og verkrunan O

Það geta oft verið fleiri en ein verkruna með sama fjölda verka

Almennt form sannana

- Þegar við ætlum að sanna að gráðugt reiknirit finni bestu laust er skipulagið oft svipað:
  - Gera ráð fyrir að til sé besta (*optimal*) lausn sem er ólík gráðugu lausninni
  - Finna "fyrsta" staðinn þar sem lausnirnar eru ólíkar
  - Sýna að við getum skipt út besta valinu fyrir gráðuga valið, án þess að lausnin verði verri

lausnin verður ekki endilega betri heldur

Sýnir að einhver besta lausn innihaldi alla gráðugu lausnina og er þess vegna jöfn gráðugu lausninni

Fyrirlestraæfingar

1. Ef allar skrárnar eru jafnstórar ($L[1] = L[2] = ... = L[n]$) en lestrartíðnir þeirra ($F[1..n]$) eru ólíkar, hvaða röð skráanna gefur lægsta heildarkostnað?

2. Gefnar eru 3 skrá með stærðum $L = [50, 20, 30]$ og lestrartíðnir $F = [50, 5, 15]$. Hvaða röð skráanna lágmarkar heildarlestrarkostnað?

3. Í stað þess að velja alltaf námskeiðið með lægsta lokatímann sem passar, þá veljum við nú alltaf námskeiðið með hæsta upphafstímann sem passar. Finnur þessi aðferð alltaf bestu lausn?