

Greining reiknirita vor 2023

Heimaverkefni 4

Skila skal þessu verkefni á vefnum [Gradescope](#).

Gradescope tekur við .pdf skjölum. Frágangur á þeim skiptir máli.

Telji nemandi að mistök hafi verið gerð við yfirferð skal tilkynna slíkt á Gradescope.

Skilafrestur er til kl. 22:00 mánudaginn 6. febrúar. Gangi þér vel!

"It is totally normal to feel confused the first time you see dynamic programming. Confusion should not discourage you. It does not represent an intellectual failure on your part, only an opportunity to get even smarter."

- Tim Roughgarden, Algorithms Illuminated

Í þessu heimaverkefni á að nota **kvika bestun** við lausn dæma. Í því felst að lýsa

- i) undirverkefnum í orðum.
- ii) rakningarvenslum og rökstyðja hvernig þau eru fengin (hafið í huga ákvörðun/ágiskun sem tekin er í hverju skrefi).
- iii) hvernig hægt er að koma í veg fyrir endurtekna útreikninga með því að geyma milliniðurstöður í viðeigandi gagnagrind.
- iv) röð undirverkefna.
- v) grunntilviki.
- vi) hvernig lausn á upphaflegu verkefni er fengin.
- vii) tímaflækju og rökstyðja hvernig hún er fengin.

1. Atvinnuviðtalið (20 stig)

Athugið dæmi 3a) á bls. 124 í Algorithms bókinni ("sum of contiguous subarray").

- a) Leysið dæmið án þess að nota kvika bestun (engin pressa!) Hver er keyrslutími reikniritsins?
- b) Leysið dæmið með kvikri bestun. Notið ramman hér á undan til að lýsa reikniritinu (atriði i – vii).

Ekki skila forritum án rökstuðnings!

2. Bestun á prófi (20 stig)

(Próf 2021) Á prófi í Finnbogastaðaskóla eru n dæmi. Fyrir dæmi i eru gefin s_i stig en reglur prófsins eru þannig að ef dæmi i er leyst þá fást engin stig fyrir dæmi $i - 1$ og $i + 1$.

Dæmi: Tilvik með $n = 4$.

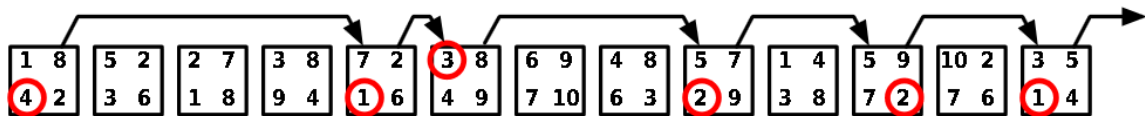
i	1	2	3	4	Stig
s_i	5	4	2	3	
Leyst		x		x	$4 + 3 = 7$

Setjið fram kvikt bestunarreiknirit sem keyrir í $O(n)$ tíma til að ákvarða hvaða dæmi eigi að leysa þannig að fjöldi stiga á prófinu sé hámarkaður. Notið ramman fyrir kvika bestun hér á undan til að lýsa reikniritinu (atriði i – vii).

3. Soby Melene (40 stig)

Soby Melene er einmenningspíll (e. solitaire) sem er spilað á n reitum sem liggja í röð. Í hverjum reit eru fjórar jákvæðar heiltölur. Leikandinn byrjar með því að setja leikpeð (e. token) á reitinn sem er lengst til vinstri. Í hverjum leik velur hann eina af tölunum fjórum sem eru á reitnum sem leikpeðið er á og færir síðan leikpeðið jafnmarga reiti til hægri og talan segir til um. Leiknum lýkur þegar leikpeðið fer framhjá reitnum sem er lengst til hægri. Markmið leiksins er að framkvæma eins margar færslur og mögulegt er.

Dæmi: Reitir sem leyfa 6 færslur (þetta er ekki mesti fjöldi mögulegra færslna):



- Sýnið að sú leikaðferð að velja lægstu töluna hverju sinni¹ skili ekki alltaf bestu lausn, þ.e. mesta möguleg fjöldi skrefa.
- Setjið fram reiknirit sem byggir á kvikri bestun til að finna lengstu mögulegu runu leikja. Notið ramman (atriði i – vii hér á undan) til að lýsa reikniritinu. Inntak í reikniritið er $n \times 4$ fylki S sem skilgreinir tölur á reitunum.
- Útfærið reikniritið úr b) í Python. Notið það til að finna lausn á dæminu. Dæmið er að finna í `example_puzzle.txt`.
- Breytið forritinu ykkar þannig að það skili, til viðbótar við lengd lengstu leikjarunu, röð leikja sem svara til lengstu leikjarununnar (nóg að skila einni röð ef það eru fleiri en ein jafnlangar). Hvert er úttakið úr forritinu ykkar þegar þið keyrið það á leikborðinu í `test_puzzle.txt`?

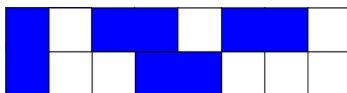
¹Þetta er dæmi um gráðugt (e. greedy) reiknirit. Slík reiknirit gefa sjaldan bestu lausn.

Þið getið notað eftirfarandi kóða til að lesa skrána inn í lista-af-listum².

```
def read_puzzle(fname):
    A=[]
    with open(fname, 'rt') as f:
        for sq in f:
            A.append([int(elt.strip()) for elt in sq.split(',')])
    return A
```

4. Fjársjóðsleit (20 stig)

Fjársjóðsleit er einmenningsspil sem er spilað á $2 \times n$ reita borði. Leikandinn raðar 2×1 kubbum á borðið þannig að kubbarnar eru annaðhvort láréttir eða lóðréttir og hver reitur á borðinu er í mesta lagi þakinn af einum kubb.



Á hverjum reit er heiltala sem getur verið bæði jákvæð eða neikvæð og táknar hún hversu mikið maður græðir eða tapar á að þekja viðkomandi reit. Það má setja eins marga kubba og maður vill á borðið. Stigin í leiknum eru summa allra gilda í reitum sem eru þakktir af kubbum.

Setjið fram reiknirit sem byggir á kvikri bestun sem ákvarðar hvar á að setja kubba þannig að stigin verði sem flest. Notið ramman (atriði i – vii hér á undan) til að lýsa reikniritinu.

²Það mætti líka nota numpy fylki.