Greining reiknirita vor 2023
Heimaverkefni 3

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 30. janúar. Gangi þér vel!

1. Rakningarvensl (30 stig)

Notið endurkvæmnistré til að leysa rakningarvenslin hér að neðan. Teiknið mynd af viðkomandi trjám og setjið lausnir fram með stóra-O rithætti.

a) $T(n) = 3T(n-1) + O(n)$

b) $T(n) = 5T(n/3) + O(n)$

c) $T(n) = 2T(n/2) + \sqrt{n}$

Ábendingar:

i) Ef $O(f(n))$ kemur fyrir í rakningarvenslum setjið þá $cf(n)$ inn í staðinn ($c > 0$ er einhver fasti).

ii) $\sum_{i=1}^{n} i = 1 + 2 + \ldots + n = n(n+1)/2$.

iii) Summa kvótaraðar (e. geometric series) er

$$\sum_{k=0}^{n} r^k = \begin{cases} \frac{r^{n+1}-1}{r-1} & r \neq 1 \\ (n+1) & r = 1. \end{cases}$$

iv) Þegar $|r| < 1$ í summunni hér að ofan þá gildir ennfremur

$$\sum_{k=0}^{\infty} r^k = \frac{1}{1-r}.$$

v) Reiknireglur fyrir veldi og logra[1] koma að góðum notum.

[1] Sjá t.d. glærur 10.1

2. Samhverfur (20 stig)

Strengur er samhverfur (e. palindrome) ef hann er eins þegar hann er lesinn afturábak og áfram. Dæmi um slíka strengi eru t.d. `ABBA` og `ADDIKALLARALLAKIDDA` (Addi kallar alla Kidda[2]).

Sérhverjum streng er hægt að skipta upp í runu af samhverfum strengjum. T.d. er hægt að skipta strengnum `BUBBASEESABANANA` ("Bubba sees a banana") í samhverfa strengi á marga vegu, þar á meðal

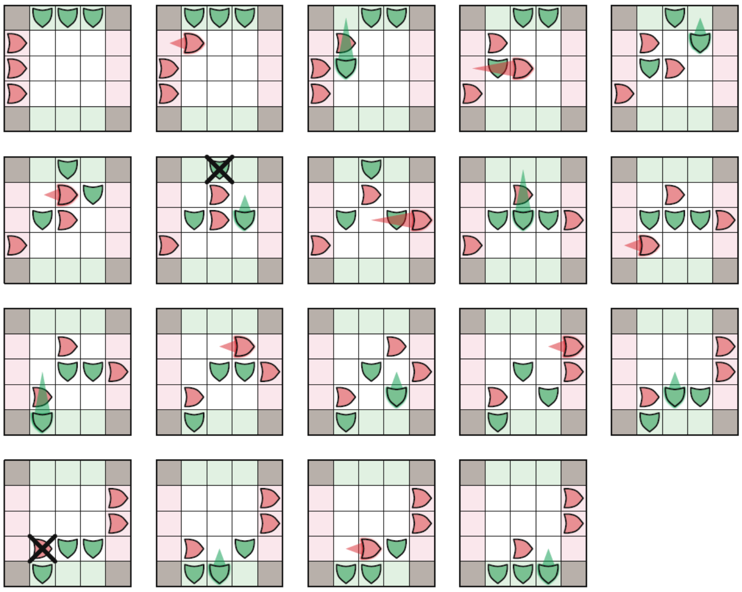
```
BUB + BASEESAB + ANANA
B + U + BB + ASEESA + B + ANANA
BUB + B + A + SEES + ABA + N + ANA
B + U + BB + A + S + EE + S + A + B + A + NAN + A
B + U + B + B + A + S + E + E + S + A + B + A + N + A + N + A
```

Skrifið endurkvæmt Python fall `split_into_palindromes(s, k)` sem kannar hvort hægt sé að skipta strengnum s upp í k samhverfa strengi, og ef svo er, þá skrifar það út strengina. Þegar forritið er t.d. keyrt á `BUBBASEESABANANA` með $k = 3$ ætti það að skrifa út `BUB BASEESAB ANANA`. Sýnið úttak úr forritinu ykkar fyrir strenginn fyrir $k = 2, \ldots, 6$. *Ábending*: Það getur borgað sig að prófa forritið á einföldum dæmum til að byrja með.

[2] `https://www.baggalutur.is/samhverfur.php`

3. Leikjatré (50 stig)

Þetta dæmi snýst um leikinn sem fjallað er um í grein 2.2 í kennslubók. Hér á að forrita spilara sem leikur alltaf besta mögulega leik í hverri stöðu, að því gefnu að slíkur leikur sé til.



Til að hjálpa ykkur af stað í verkefninu þá fylgir því Python kóði sem

- heldur utan um stöðu á leikborðinu.
- framkvæmir færslur á peðum.
- teiknar ”mynd” af borðinu (úff!)
- finnur hvort annar spilarinn sé búinn að vinna.
- útfærir útgáfu af leiknum fyrir tvo mennska spilara.

Borðið er geymt sem listi-af-listum[3] þar sem $T[0]$ er n staka listi sem geymir staðsetningar peða fyrri leikmannsins og $T[1]$ er tilsvarandi listi fyrir seinni leikmanninn.

Verkefni ykkar er að útfæra reikniritið bls. 76 og nota það til að ákvarða leiki fyrir annan leikmanninn. Kóðinn sem fylgir verkefninu ræður við hvaða stærð af borði sem er, en vegna þess að leikjatréið vex mjög hratt með stærð borðsins er líklegt að þið séuð takmörkuð í leitinni við 3×3 borð eins og það sem sýnt er á myndinni hér að ofan[4].

Þið eigið að skila kóða fyrir `FindGoodMove` fallið sem finnur besta leik í gefinni stöðu auk kóðans sem heldur utan um leikinn sjálfann (”game loop”). Sýnið einnig skjáskot úr leiknum.

Það er ljóst að seinni leikmaðurinn vinnur alltaf þegar $n = 1$. Gildir það sama um $n = 2$ og $n = 3$? Setjið fram tilgátu um hvor getur alltaf sigrað á $n > 3$ borði?

[3]Þetta er óhagkvæm leið til að geyma stöðu borðsins hvað keyrslutíma varðar en fyrsta mál á dagskrá er að fá reikniritið til að virka.

[4]$n \times n$ borð svarar til n peða hjá hvorum spilara.