

HEIMADÆMI 3

TÖL203G Tölvunarfræði 2¹

Kári Hlynsson

Háskóli Íslands

2. febrúar 2023

Verkefni 1

Breytið `FourSum.java` í `FourSumFast.java` á sama hátt og er gert með `ThreeSumFast.java`. Skilið kóða fallsins `count` (sem texta, ekki skjáskoti) og skjáskoti af keyrslu `FourSum` og `FourSumFast` á gagnaskránni `1Kints.txt`. Þá eiga að finnast 13654 ferndir. Athugið að þið þurfið að laga kóðann aðeins, því `texttttFourSum.java` notar `long` fylki í stað `int` fylkis og innlestur gagnanna er aðeins ólíkur.

Lausn

Breytingarnar eru lítillgar og sjást fyrir neðan í forriti 1.

```
49     public static int count(long[] a) {
50         int N = a.length;
51         Arrays.sort(a);
52         int cnt = 0;
53         if (containsDuplicates(a))
54             throw new IllegalArgumentException("array contains duplicates");
55         for (int i = 0; i < N; i++) {
56             for (int j = i + 1; j < N; j++) {
57                 for (int k = j + 1; k < N; k++) {
58                     int l = Arrays.binarySearch(a, - (a[i] + a[j] + a[k]));
59                     if (l > k) cnt++;
60                 }
61             }
62         }
63         return cnt;
64     }
65 }
```

Forrit 1: Fallið `count` í `FourSumFast.java`

Mynd 1 sýnir keyrslu í skel á `FourSum.java` og síðan `FourSumFast.java`.

¹Slóð á Github source kóða:

```
*[master] [~/Github/TOL203G/HD3/src]$ java FourSum < 1Kints.txt
13654
*[master] [~/Github/TOL203G/HD3/src]$ java FourSumFast < 1Kints.txt
13654
*[master] [~/Github/TOL203G/HD3/src]$ █

[0] 0:nvim- 1:zsh* "vr2t-dw200.rhi.hi.is" 11:21 02-Feb-23
```

Mynd 1: Keyrsla FourSum og FourSumFast í skel

Eins og má sjá fáum við 13654 sem er viðbúinn fjöldi fernda.

Verkefni 2

Framhald af æfingadæminu að ofan.

- (a) Finnið raunhæf neðri mörk á vaxtarhraða keyrslutíma reikniritis sem leysir þetta verkefni, þ.e. hversu margar aðgerðir þurfa öll reiknirit að nota til þess að leysa þetta verkefni (sem fall af N)? Rökstyðjið svarið í nokkrum orðum.
- (b) Það er hægt að leysa verkefnið á mun hraðvirkar hátt en gert var í æfingadæminu, með því að nýta sér fyrri útreikninga í B . Hugmyndin er þið eruð að reikna út $B[i, j]$ þá eruð þið nýbúin að reikna út $B[i, j-1]$. Er ekki hægt að nota það gildi? Útfærið þetta reiknirit í Java og keyrið það fyrir sömu gildi á N og gert var í æfingadæminu. Hver er vaxtarhraði þessa nýja reikniritis? Skilið kóðanum (sem texta, ekki skjáskoti) og svarinu.

Lausn

Hluti (a)

Látum $\sigma(i, j) := \sum_{k=i}^j a_k$. Við getum sett upp töflu sem sýnir hvernig fylkið B lítur út fyrir gefna inntaksstærð N :

$\downarrow i \rightarrow j$	1	2	...	N
1	$\sigma(1, 1)$	—	...	—
2	$\sigma(2, 1)$	$\sigma(2, 2)$...	—
\vdots	\vdots	\vdots	\ddots	\vdots
N	$\sigma(N, 1)$	$\sigma(N, 2)$...	$\sigma(N, N)$

Tafla 1: Útlit fylkisins B .

Við miðum almenna kostanaðarlíkanið út frá fjölda fallakalla á $\sigma(i, j)$. Við sjáum að heildafjöldi kalla er $1 + 2 + \dots + N$ svo við fáum

$$T(N) = 1 + 2 + \dots + N = \frac{N(N+1)}{2} \sim \frac{1}{2}N^2$$

m.ö.o. er $T(N) \sim \Omega(N^2)$.

Hluti (b)

Við skulum hefja umfjöllunina á upprunalega fallinu.