

## Æfingadæmi 1, Hlutsummur.java

```
19 // Búa til slembigildi í fylkið (nota t.d. StdRandom.uniformInt())
20 // FYLLA INN ...
21 for (int i = 0; i < N; i++) {
22     A[i] = StdRandom.uniformInt(N);
23 }
24
25 // Taka tímann á útreikningi
26 Stopwatch timer = new Stopwatch();
27 int[][] B = reiknaHS(A);
28 double eTime = timer.elapsedTime();
29
30 StdOut.println("Elapsed time = " + eTime);
31
32 // Einföld stikkprufa til að athuga hvort rétt reiknað ...
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Nám\Vor25\TÖL203\Heimadæmi\Heimadæmi3> javac Hlutsummur.java
PS C:\Nám\Vor25\TÖL203\Heimadæmi\Heimadæmi3> java Hlutsummur 50
Elapsed time = 0.0
A[2] + A[3] = 21
B[2][3] = 0
PS C:\Nám\Vor25\TÖL203\Heimadæmi\Heimadæmi3> java Hlutsummur 12345
Elapsed time = 0.313
A[2] + A[3] = 15510
B[2][3] = 0
PS C:\Nám\Vor25\TÖL203\Heimadæmi\Heimadæmi3>
```

Highlightaða for loopan er það eina sem ég bætti við. Keyrði svo tvisvar með tvö mismunandi input. 12345 tóluvert stærra en bara 50. 12345 tók 0,313sek meðan 50 tók heilar 0,0 sek. Líklegast margar tölur þarna eftirá sem prentast ekki út frá gefnum kóða.

Dæmi 1, sem er framhald af æfingadæminu (sem er ástæðan fyrir því að ég ákvað að leyfa æfingadæminu að fylgja með)

- Útreikningar fela í sér að reikna allar summur fylkisins. Fylki með lengd  $N$  hefur  $(N(N+1))/2$  fjölda undirfylkja og þá þarf hvaða reiknirit sem reynir við verkefnið að skoða  $N^2$  samlagningar. Þá eru neðri mörkin  $\Omega(N^2)$  (annað font, copyað úr glærunum).
- Með því að endurnýta summuna á undan er hægt að spara tíma. Ef ég reikna  $1+2$ , þá þarf ég næst ekki að reikna  $1+2+3$  því ég er búinn að reikna  $1+2$ . Ég reikna bara  $3+3$ . Spara mér heilt +. Það gerir vaxtargráðuna  $\Theta(N^2)$  (aftur copyað úr glærunum).

```
12 // FYLLA INN ...
13 for (int i = 0; i < N; i++) {
14     B[i][i] = A[i];
15     for (int j = i + 1; j < N; j++) {
16         B[i][j] = B[i][j - 1] + A[j]; // Fyrira notað til að reikna seinna
17     }
18 }
19 return B;
20
Run | Debug
21 public static void main(String[] args) {
22     int N = Integer.parseInt(args[0]);
23     int[] A = new int[N];
24
25     // Búa til slembigildi í fylkið (nota t.d. StdRandom.uniformInt())
26     // FYLLA INN ...
27     /*for (int i = 0; i < N; i++) {
28         A[i] = StdRandom.uniformInt(N);
29     }*/
30 }
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Nám\Vor25\TÖL203\Heimadæmi\Heimadæmi3> javac Hlutsummur.java
PS C:\Nám\Vor25\TÖL203\Heimadæmi\Heimadæmi3> java Hlutsummur 12345
Elapsed time = 0.34
A[2] + A[3] = 0
B[2][3] = 0
PS C:\Nám\Vor25\TÖL203\Heimadæmi\Heimadæmi3> java Hlutsummur 12345
Elapsed time = 0.357
A[2] + A[3] = 0
B[2][3] = 0
PS C:\Nám\Vor25\TÖL203\Heimadæmi\Heimadæmi3>
```

## Dæmi 2.

- Þá nota ég bara binary search/helmingunarleit. Tek lyftuna á  $M/2$ , og ef hann brotnar, þá fer ég niður í  $(M/2)/2$  o.s.frv. en ef hann brotnar ekki þá fer ég upp á  $M/2 * 1.5$  hæð og vinn mig þaðan.
- Sko. Byrja sirka í miðjunni og ef hann brotnar ekki, þá fer ég upp um þó nokkrar hæðir o.s.frv. þar til hann brotnar. Þegar hann brotnar, þá fer ég niður um helminginn af mismuninum. Segjum ég slepti seinast af 50. hæð og hann brotnar svo af 70. hæð, þá fer ég næst niður í 60. hæð. Mjög svipað binary search, nema núna er ég varkárri og er ekki alltaf að helminga upp eða niður þótt ég viti hvað  $M$  er

## Dæmi 3.

Annars staðar í kóðanum er ég með `int N = 5;`.

Fyrsta lykkja gefur útkomuna 15, því  $5+4+3+2+1 = 15$  og það eru 5 keyrslur.  $\Theta(N)$  hraði.

Önnur lykkja gefur útkomuna 30, því  $0+1+4+9+16 = 30$  og það eru einnig 5 keyrslur, en bara því  $N = 5$ . Ástæðan fyrir 0, 1, 4, 9, og 16 er  $i*i$ .  $0*0$ ,  $1*1$ ,  $2*2$ ,  $3*3$ , og  $4*4$ .  $i$  og  $j$  eru ++, svo þau hækka samhliða um einn per ítrekun.  $\Theta(N^2)$  hraði því loopan er tvöföld.

Þriðja lykkja gefur útkomuna 9, því  $3*3$  er 9. 2 keyrslur. 2 keyrslur því innri og ytri lykkjan eru nákvæmlega eins, fyrir utan  $i$  og  $j$ .  $i$  og  $j$  eru bæði  $< N$ , 5 í þessu tilfelli. Aftasti liður for loopunar eru  $i \leq j$  sem setur sig sjálft ítrekað í annað veldi. Þess vegna eru tölurnar  $1 + 2 + 4 + 8$  en ekkert meir. 5ta keyrsla hefði verið 16, en  $i > j < N$ . Báðar lykkjur keyra á  $\log N$  en þar sem þær tvöfaldast hverja ítrun er það  $\Theta(\log^2 N)$

```
Lykkjan keyrði 5 sinnum.  
Lykkjan keyrði 30 sinnum.  
Lykkjan keyrði 9 sinnum.  
15  
30  
9  
PS C:\Nám\Vor25\TÖL203\Heimadæmi\Heimadæmi3>
```

```
a. public static int loop1(int N) {  
b.     int s = 0;  
c.     int counter = 0;  
d.     for (int i=N; i>0; i--) {  
e.         s+=i;  
f.         counter++;  
g.     }  
h.     System.out.println("Lykkjan keyrði " + counter + " sinnum.");
```

```
i.         return s;
j.     }
k.
l.     public static int loop2(int N) {
m.         int s = 0;
n.         int counter = 0;
o.         for (int i=0; i<N; i++) {
p.             for (int j=0; j<i*i; j++) {
q.                 s++;
r.                 counter++;
s.             }
t.         }
u.         System.out.println("Lykkjan keyrði " + counter + " sinnum.");
v.         return s;
w.     }
x.
y.     public static int loop3(int N) {
z.         int s = 0;
aa.        int counter = 0;
bb.        for (int i=1; i<N; i*=2) {
cc.            for (int j=1; j<N; j*=2) {
dd.                s++;
ee.                counter++;
ff.            }
gg.        }
hh.        System.out.println("Lykkjan keyrði " + counter + " sinnum.");
ii.        return s;
jj.    }
```

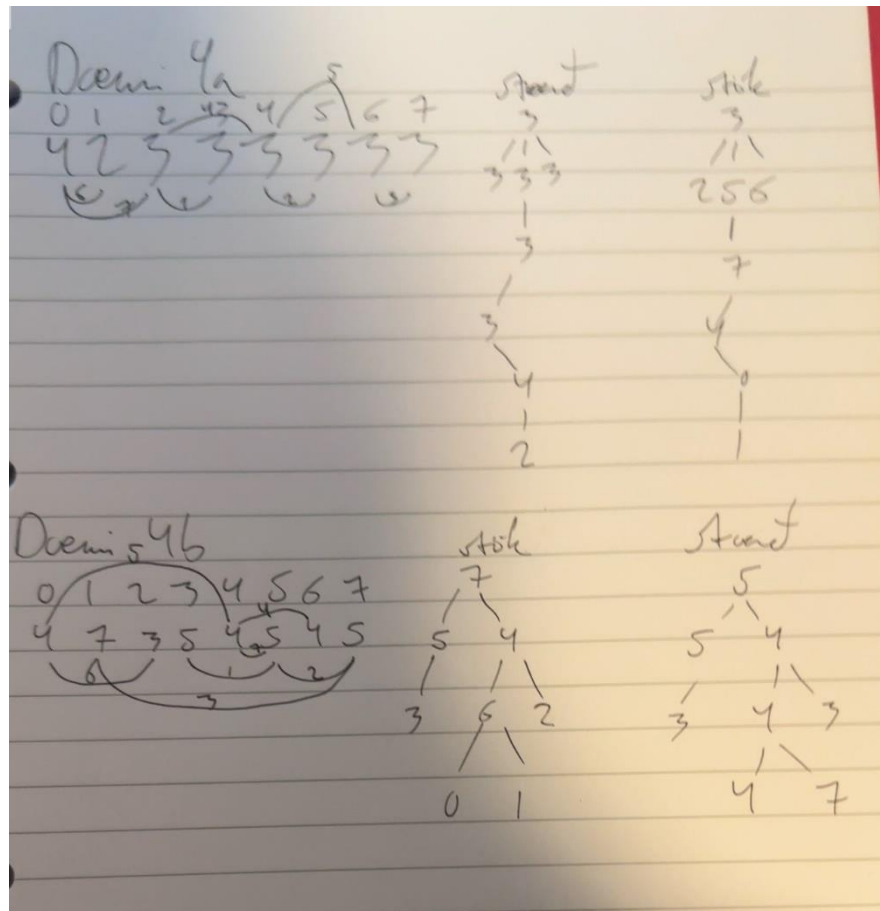
#### Dæmi 4.

a. Fylkið  $[4, 2, 3, 3, 3, 3, 3, 3]$  getur virkað.

1. Tengi stak 2 við 3
2. Tengi stak 4 við 5
3. Tengi stak 6 við 7
4. Tengi 4 við 6, til að sameina
5. Tengi 2 við 4, til að sameina
6. Tengi stak 0 við 1
7. Tengi stak 0 við 2

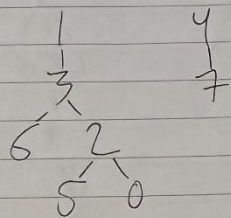
b. Fylkið  $[4, 7, 3, 5, 4, 5, 4, 5]$  getur einnig virkað.

1. Tengi stak 3 við 5
2. Tengi 7 við 5, til að sameina
3. Tengi 1 við 7, til að sameina
4. Tengi stak 4 við 6
5. Tengi 0 við 4, til að sameina
6. Tengi 2 við 0, til að sameina
7. Tengi 5 við 4, til að sameina



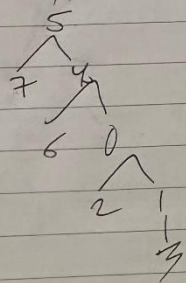
Dæmi 5 a,b, og c

Dæmi 5a



Dæmi 5b

$(0,2), (1,3), (4,6), (5,7), (0,1), (4,3), (5,1)$



Dæmi 5c

$(0,1)(1,2)(2,3)(3,4)(4,5)(5,6)(6,7)$

