Dæmatímaverkefni 3 - USTY

Harpa Guðjónsdóttir

4.febrúar 2016

Atriði 1

For lykkja keyrð 30 sinnum, eða jafnt oft og breytan NUMBER_OF_PROBLEMS segir til um.

```
for(int i=0; i<NUMBER_OF_PROBLEMS; i++){
    Solver.findAndPrintSolution(Problematic.nextProblem());
    }
```

Forritið keyrir í röð og úttakið er alltaf eins, ekki er kallað á næsta tilvik fyrr en tilvikið á undan hefur skilað. Þetta skilar:

```
Processors: 4
Solutions:
Problem: (557,160) - (110,177) - length: 464 - iterations: 217040
Problem: (461,73) - (99,155) - length: 444 - iterations: 204752
Problem: (615,254) - (394,470) - length: 437 - iterations: 188783
Problem: (296,109) - (34,279) - length: 432 - iterations: 243092
Problem: (85,493) - (61,440) - length: 77 - iterations: 11811
Problem: (263,29) - (336,608) - length: 652 - iterations: 329093
Problem: (279,152) - (618,465) - length: 652 - iterations: 383761
Problem: (276,214) - (2,334) - length: 394 - iterations: 263061
Problem: (113,34) - (489,423) - length: 765 - iterations: 342223
Problem: (60,307) - (350,354) - length: 337 - iterations: 149614
Problem: (393,490) - (64,8) - length: 811 - iterations: 406964
Problem: (493,167) - (222,265) - length: 369 - iterations: 183582
Problem: (422,584) - (107,589) - length: 320 - iterations: 125153
Problem: (216,64) - (525,344) - length: 589 - iterations: 304957
Problem: (626,79) - (148,152) - length: 551 - iterations: 200691
Problem: (303,611) - (328,206) - length: 430 - iterations: 191063
Problem: (313,267) - (603,279) - length: 302 - iterations: 181243
```

```
Problem: (232,148) - (381,22)  - length: 275 - iterations: 133651
Problem: (423,194) - (398,320) - length: 151 - iterations: 45554
Problem: (119,429) - (111,580) - length: 159 - iterations: 48948
Problem: (13,309)  - (113,149) - length: 260 - iterations: 74511
Problem: (279,35)  - (623,21)  - length: 358 - iterations: 147055
Problem: (413,605) - (248,223) - length: 547 - iterations: 260947
Problem: (130,454) - (548,561) - length: 525 - iterations: 296899
Problem: (297,432) - (392,501) - length: 164 - iterations: 53931
Problem: (436,382) - (324,618) - length: 348 - iterations: 213150
Problem: (359,475) - (441,372) - length: 185 - iterations: 68236
Problem: (299,273) - (344,252) - length: 66  - iterations: 8754
Problem: (431,263) - (150,389) - length: 407 - iterations: 270185
Problem: (553,90)  - (379,333) - length: 417 - iterations: 160783
All done
Total time: 23941 ms
```

Atriði 2

Forritið keyrir tilvikin öll í einu á mismunandi þráður. Nýr þráður er búinn
til fyrir hvert tilvik.

```
    Thread[] threads = new Thread[NUMBER_OF_PROBLEMS];
    for(int i = 0; i < NUMBER_OF_PROBLEMS;i++){
        Problem newProblem = Problematic.nextProblem();
        threads[i] = new Thread(makeNewRunnable(i,newProblem));
        threads[i].start();
    }

    try {
     for(int i = 0; i<NUMBER_OF_PROBLEMS; i++){
     threads[i].join();
         }
} catch (InterruptedException e) {
// TODO Auto-generated catch block
e.printStackTrace();
}
```

Forritið keyrir á nýjum þráð fyrir hvert tilvik, úttakið er ekki alltaf eins en
þó alltaf mjög svipað. Það sem gerist er að Solver.findAndPrintSolution() er
fljótara að keyra ef lenght er minni og þ.l. iterations færri. Úttakið er því

í nokkurs konar vaxandi röð þó svo að stundum geti það munað einhverju eins og sjá má með línu 3 og 4 í úttakinu hér að neðan þar sem length 77 klárar á undan 66. Þetta gerist hinsvegar ekki alltaf og oft klárar problem af lengd 66 á undan 77.

Dæmi um úttak:

```
Processors: 4
Solutions:
Problem: (85,493) - (61,440) - length: 77 - iterations: 11811
Problem: (299,273) - (344,252) - length: 66 - iterations: 8754
Problem: (423,194) - (398,320) - length: 151 - iterations: 45554
Problem: (119,429) - (111,580) - length: 159 - iterations: 48948
Problem: (359,475) - (441,372) - length: 185 - iterations: 68236
Problem: (13,309) - (113,149) - length: 260 - iterations: 74511
Problem: (297,432) - (392,501) - length: 164 - iterations: 53931
Problem: (60,307) - (350,354) - length: 337 - iterations: 149614
Problem: (422,584) - (107,589) - length: 320 - iterations: 125153
Problem: (553,90) - (379,333) - length: 417 - iterations: 160783
Problem: (232,148) - (381,22) - length: 275 - iterations: 133651
Problem: (279,35) - (623,21) - length: 358 - iterations: 147055
Problem: (493,167) - (222,265) - length: 369 - iterations: 183582
Problem: (615,254) - (394,470) - length: 437 - iterations: 188783
Problem: (303,611) - (328,206) - length: 430 - iterations: 191063
Problem: (436,382) - (324,618) - length: 348 - iterations: 213150
Problem: (313,267) - (603,279) - length: 302 - iterations: 181243
Problem: (461,73) - (99,155) - length: 444 - iterations: 204752
Problem: (557,160) - (110,177) - length: 464 - iterations: 217040
Problem: (276,214) - (2,334) - length: 394 - iterations: 263061
Problem: (296,109) - (34,279) - length: 432 - iterations: 243092
Problem: (431,263) - (150,389) - length: 407 - iterations: 270185
Problem: (626,79) - (148,152) - length: 551 - iterations: 200691
Problem: (130,454) - (548,561) - length: 525 - iterations: 296899
Problem: (413,605) - (248,223) - length: 547 - iterations: 260947
Problem: (113,34) - (489,423) - length: 765 - iterations: 342223
Problem: (216,64) - (525,344) - length: 589 - iterations: 304957
Problem: (263,29) - (336,608) - length: 652 - iterations: 329093
Problem: (279,152) - (618,465) - length: 652 - iterations: 383761
Problem: (393,490) - (64,8) - length: 811 - iterations: 406964
All done
```

```
Total time: 4447 ms
```

Atriði 3

Forritið keyrir ákveðin fjölda tilvika í einu, skoðum það að keyra 5 tilvik í einu. Þræðir eru settir af stað í gegnum thread pool. Nýtt tilvik er ekki sett af stað fyrr en þráður losnar í thread poolinu.

```
ExecutorService threadPool = Executors.newFixedThreadPool(POOL_SIZE);

    for(int i = 0; i < NUMBER_OF_PROBLEMS;i++){
     Problem newProblem = Problematic.nextProblem();
     Runnable worker = makeNewRunnable(i,newProblem);
     threadPool.execute(worker);
    }
    try {
     threadPool.shutdown();
     threadPool.awaitTermination(5, TimeUnit.MINUTES);
    } catch (InterruptedException e) {
     e.printStackTrace();
    }
```

Að auki skilgreindi ég breytu fyrir utan main fallið, POOL_SIZE, beint fyrir neðan skilgreininguna á NUMBER_OF_PROBLEMS

```
private static final int POOL_SIZE = 5;
```

Það sem gerist í þessu er að fyrstu 5 problemin(ef thread pool size er af stærð 5) í atriði 1 klára í stærðar röð(eða því sem næst), síðan þegar fyrstu klárast fer næsti af stað og þræðirnir klárast nokkurn vegin innbyrðist í stærðar röð nema t.d. þegar þráður er búin að vera lengi að keyra og síðan kemur nýtt tilvik sem er styttra inn þá klárar þráðurinn sem er lengi að keyra kannski á undan því hann var búin að vera lengi í gangi áður en stutta problemið kom inn.Sama mundi gerast ef thread pool stærðin væri önnur, t.d. x, en þá raðast fyrstu x problemin í artriði 1 í klára í stærðar röð(eða því sem næst) Msimunandi thread pool stærð skilar mismunandi niðurstöðu, til dæmis ef notast er við 30 samhliða þræði þá skilar það svipaðri niðurstöðu og ef forritið keyrir hvert tilvik á nýjum þræði(atvik 2).
Dæmi um úttak(með thread pool af stærð 5)

```
Processors: 4
```

```
Solutions:
Problem: (85,493) - (61,440) - length: 77 - iterations: 11811
Problem: (557,160) - (110,177) - length: 464 - iterations: 217040
Problem: (615,254) - (394,470) - length: 437 - iterations: 188783
Problem: (461,73) - (99,155) - length: 444 - iterations: 204752
Problem: (296,109) - (34,279) - length: 432 - iterations: 243092
Problem: (263,29) - (336,608) - length: 652 - iterations: 329093
Problem: (60,307) - (350,354) - length: 337 - iterations: 149614
Problem: (276,214) - (2,334) - length: 394 - iterations: 263061
Problem: (113,34) - (489,423) - length: 765 - iterations: 342223
Problem: (493,167) - (222,265) - length: 369 - iterations: 183582
Problem: (279,152) - (618,465) - length: 652 - iterations: 383761
Problem: (422,584) - (107,589) - length: 320 - iterations: 125153
Problem: (303,611) - (328,206) - length: 430 - iterations: 191063
Problem: (626,79) - (148,152) - length: 551 - iterations: 200691
Problem: (313,267) - (603,279) - length: 302 - iterations: 181243
Problem: (393,490) - (64,8) - length: 811 - iterations: 406964
Problem: (423,194) - (398,320) - length: 151 - iterations: 45554
Problem: (119,429) - (111,580) - length: 159 - iterations: 48948
Problem: (13,309) - (113,149) - length: 260 - iterations: 74511
Problem: (216,64) - (525,344) - length: 589 - iterations: 304957
Problem: (232,148) - (381,22) - length: 275 - iterations: 133651
Problem: (279,35) - (623,21) - length: 358 - iterations: 147055
Problem: (297,432) - (392,501) - length: 164 - iterations: 53931
Problem: (299,273) - (344,252) - length: 66 - iterations: 8754
Problem: (359,475) - (441,372) - length: 185 - iterations: 68236
Problem: (413,605) - (248,223) - length: 547 - iterations: 260947
Problem: (436,382) - (324,618) - length: 348 - iterations: 213150
Problem: (553,90) - (379,333) - length: 417 - iterations: 160783
Problem: (130,454) - (548,561) - length: 525 - iterations: 296899
Problem: (431,263) - (150,389) - length: 407 - iterations: 270185
All done
Total time: 6021 ms
```

Runnable fall

Einnig var skilgreint runnable fall, sem notast var við í atriði 2 og 3

```
private static Runnable makeNewRunnable(final int number, final Problem problem){
    return new Runnable(){
```

```
    @Override
    public void run() {
    Solver.findAndPrintSolution(problem);
}
    };
}
```