

Við áttum ekkert við Problem.java, Problematic.java, Tile.java og Solver.java.

Við bjuggum til klasann ProblemRunner til að geta búið til nýtt tilvik af Thread. Í ProblemRunner klasanum er fallið run, en í því er kallað á findAndPrintSolution úr Solver klasanum (sem leysir problemið, en hvert problem tekur mismunandi langan tíma að leysa) og sent inn tilvik af Problem og númeri þess, en við bættum nr breytunni inn til að geta séð hvaða problem var í gangi hverju sinni.

Í ThreadAssignment01Main klasanum bjuggum til breytuna int RUN_SETTINGS = 1/2/3; til að það sé hægt að vita hvar í runferlinu hvert problem er? Þ.e. 1 til að uppfylla skilyrðið í lið 1, 2 til að uppfylla skilyrðið í lið 2 og 3 til að uppfylla skilyrðið í 3. Það þarf sem sagt að breyta þessari tölu til að fá hvert og eitt "svar".

Í main fallinu byrjuðum við á að búa til fylki af problemum.

```
Problem[] problems = new Problem[NUMBER_OF_PROBLEMS];
for(int i = 0; i < NUMBER_OF_PROBLEMS; i++){
    problems[i] = Problematic.nextProblem();
}
```

Problematic.nextProblem() gefur manni nýtt random problem.

Næst komum við að if-setningum sem keyra eftir hvaða RUN_SETTING er stillt. Hver if-setning inniheldur for-lykkju sem hleypur í gegnum flykið af problemunum og keyrir svo mismunandi föll á þau.

Ef verið er að leysa fyrir part 1 af verkefninu keyrist kóðinn:

```
if(RUN_SETTINGS == 1) {
    /* Part 1:
     * Sequentially: Don't run the next instance until the one before has returned
     */
    System.out.println("Starting part 1. Running sequentially.");

    for (int i = 0; i < NUMBER_OF_PROBLEMS; i++) {
        Solver.findAndPrintSolution(problems[i], i);
    }
}
```

Solver.findAndPrintSolution tekur inn problem parameter, leysir lausnina og skilar niðurstöðu. Hvert og eitt problem tekur mis langan tíma að leysa. Við létum fallið hins vegar líka taka inn sætistölu problemsins í fylkinu til að auðvelda okkur að sjá hvaða problem væri verið að leysa hverju sinni.

Hér eru problemin bara keyrð hvert á fætur öðru. Eitt byrjar ekki nema það sem var á undan sé lokið.

Dæmi um output:

Starting part 1. Running sequentially.

```
Problem#0: (557,160) - (110,177) - length: 464 - iterations: 217040
Problem#1: (461,73) - (99,155) - length: 444 - iterations: 204752
Problem#2: (615,254) - (394,470) - length: 437 - iterations: 188783
Problem#3: (296,109) - (34,279) - length: 432 - iterations: 243092
Problem#4: (85,493) - (61,440) - length: 77 - iterations: 11811
Problem#5: (263,29) - (336,608) - length: 652 - iterations: 329093
Problem#6: (279,152) - (618,465) - length: 652 - iterations: 383761
Problem#7: (276,214) - (2,334) - length: 394 - iterations: 263061
Problem#8: (113,34) - (489,423) - length: 765 - iterations: 342223
Problem#9: (60,307) - (350,354) - length: 337 - iterations: 149614
Problem#10: (393,490) - (64,8) - length: 811 - iterations: 406964
Problem#11: (493,167) - (222,265) - length: 369 - iterations: 183582
Problem#12: (422,584) - (107,589) - length: 320 - iterations: 125153
Problem#13: (216,64) - (525,344) - length: 589 - iterations: 304957
Problem#14: (626,79) - (148,152) - length: 551 - iterations: 200691
Problem#15: (303,611) - (328,206) - length: 430 - iterations: 191063
Problem#16: (313,267) - (603,279) - length: 302 - iterations: 181243
Problem#17: (232,148) - (381,22) - length: 275 - iterations: 133651
Problem#18: (423,194) - (398,320) - length: 151 - iterations: 45554
Problem#19: (119,429) - (111,580) - length: 159 - iterations: 48948
Problem#20: (13,309) - (113,149) - length: 260 - iterations: 74511
Problem#21: (279,35) - (623,21) - length: 358 - iterations: 147055
Problem#22: (413,605) - (248,223) - length: 547 - iterations: 260947
Problem#23: (130,454) - (548,561) - length: 525 - iterations: 296899
Problem#24: (297,432) - (392,501) - length: 164 - iterations: 53931
Problem#25: (436,382) - (324,618) - length: 348 - iterations: 213150
Problem#26: (359,475) - (441,372) - length: 185 - iterations: 68236
Problem#27: (299,273) - (344,252) - length: 66 - iterations: 8754
Problem#28: (431,263) - (150,389) - length: 407 - iterations: 270185
Problem#29: (553,90) - (379,333) - length: 417 - iterations: 160783
```

Hér prentast niðurstöðurnar út í hvert á eftir öðru, eða í þeirri röð sem for-lykkjan veiddi þau úr fylkinu.

Ef verið er að leysa fyrir part 2 af verkefninu keyrist kóðinn:

```
else if(RUN_SETTINGS == 2) {
/* Part 2:
 * All at once in separate threads. A new thread is created for each instance.
 * */
    System.out.println("Starting part 2. Running all at once in separate threads.");

    for (int i = 0; i < NUMBER_OF_PROBLEMS; i++) {
        new Thread(new ProblemRunner(problems[i], i)).start();
    }
}
```

Í forlykkjunni er búinn til nýr þráður og honum startað með `new Thread().start`. Þetta er jafngilt því að gera:

```
Thread myThread = new Thread(someRunnableClass);

myThread.start();
```

Thread() tekur inn nýtt tilvik af problemRunner klasanum sem við gerðum og sögðum frá ofar í skjalinu. Thread() býr til nýjan þráð sem gerir nýtt tilvik af ProblemRunner og keyrir fallið til að leysa eitt problem. Hvert problem er sem sagt leyst í nýjum þræði og niðurstöðum skilað úr hverjum þræði fyrir sig þegar hann hefur lokið keyrslu, sem gerir að verkum að niðurstöðurnar úr problemunum prentast ekki út í þeirri röð sem þær voru "teknar úr fylkinu"

Dæmi um output:

```
Starting part 2. Running all at once in separate threads.
Problem#27: (299,273) - (344,252) - length: 66 - iterations: 8754
Problem#4: (85,493) - (61,440) - length: 77 - iterations: 11811
Problem#26: (359,475) - (441,372) - length: 185 - iterations: 68236
Problem#18: (423,194) - (398,320) - length: 151 - iterations: 45554
Problem#19: (119,429) - (111,580) - length: 159 - iterations: 48948
Problem#24: (297,432) - (392,501) - length: 164 - iterations: 53931
Problem#17: (232,148) - (381,22) - length: 275 - iterations: 133651
Problem#12: (422,584) - (107,589) - length: 320 - iterations: 125153
Problem#15: (303,611) - (328,206) - length: 430 - iterations: 191063
Problem#20: (13,309) - (113,149) - length: 260 - iterations: 74511
Problem#1: (461,73) - (99,155) - length: 444 - iterations: 204752
Problem#2: (615,254) - (394,470) - length: 437 - iterations: 188783
Problem#11: (493,167) - (222,265) - length: 369 - iterations: 183582
Problem#16: (313,267) - (603,279) - length: 302 - iterations: 181243
Problem#21: (279,35) - (623,21) - length: 358 - iterations: 147055
Problem#29: (553,90) - (379,333) - length: 417 - iterations: 160783
Problem#9: (60,307) - (350,354) - length: 337 - iterations: 149614
Problem#7: (276,214) - (2,334) - length: 394 - iterations: 263061
Problem#3: (296,109) - (34,279) - length: 432 - iterations: 243092
Problem#14: (626,79) - (148,152) - length: 551 - iterations: 200691
Problem#25: (436,382) - (324,618) - length: 348 - iterations: 213150
Problem#22: (413,605) - (248,223) - length: 547 - iterations: 260947
Problem#23: (130,454) - (548,561) - length: 525 - iterations: 296899
Problem#0: (557,160) - (110,177) - length: 464 - iterations: 217040
Problem#8: (113,34) - (489,423) - length: 765 - iterations: 342223
Problem#13: (216,64) - (525,344) - length: 589 - iterations: 304957
Problem#5: (263,29) - (336,608) - length: 652 - iterations: 329093
Problem#28: (431,263) - (150,389) - length: 407 - iterations: 270185
Problem#10: (393,490) - (64,8) - length: 811 - iterations: 406964
Problem#6: (279,152) - (618,465) - length: 652 - iterations: 383761
```

Hér prentastniðurstöðurnar út um leið og þær eru tilbúnar, þó svo að Problem#0 hafi verið sett fyrst í gang, þýðir það ekki að það skrifist fyrst út.

Ef verið er að leysa fyrir part 3 af verkefninu keyrist kóðinn:

```
else if(RUN_SETTINGS == 3) {  
    /* Part 3:  
    * A certain number at a time. Threads are run through a thread pool of a  
    * certain size (students can try different sizes). New instances aren't run  
    * until a thread is free in the thread pool.  
    */  
    System.out.println("Starting part 3. Pooling.");  
  
    ExecutorService threadPool = Executors.newFixedThreadPool(POOL_SIZE);  
  
    for (int i = 0; i < NUMBER_OF_PROBLEMS; i++) {  
        threadPool.execute(new ProblemRunner(problems[i], i));  
    }  
  
    threadPool.shutdown();  
}
```

Hér er búið til threadPool breytu, en hún keyrir POOL_SIZE marga þræði í einu. Næst er keyrt threadPool.execute() fyrir hvert og eitt problem í problem fylkinu. Nýtt tilvik af ProblemRunner(sem tekur inn problem og sætisnúmer þess í fylkinu) er sent inn á hvern threadPool.execute og þar að leiðandi eru bara keyrð POOL_SIZE problem í einu. Niðurstöðunum er svo skilað í þeirri röð sem þær koma, en næstu POOL_SIZE problemum er ekki hleypt af stað fyrirren síðasta problemið í POOL_SIZE problemunum á undan hefur skilað niðurstöðum. Ef POOL_SIZE breytan er = 5, skrifast 5 fyrstu út, svo 5 næstu eftir að þessi fyrstu hafa skrifast út o.s.frv.

Það er svo endað á að keyra .shutdown() á threadPool til að hætta að búa til nýja þræði.

Dæmi um output:

Starting part 3. Pooling.

```
Problem#4: (85,493) - (61,440) - length: 77 - iterations: 11811
Problem#2: (615,254) - (394,470) - length: 437 - iterations: 188783
Problem#3: (296,109) - (34,279) - length: 432 - iterations: 243092
Problem#0: (557,160) - (110,177) - length: 464 - iterations: 217040
Problem#1: (461,73) - (99,155) - length: 444 - iterations: 204752
Problem#7: (276,214) - (2,334) - length: 394 - iterations: 263061
Problem#5: (263,29) - (336,608) - length: 652 - iterations: 329093
Problem#8: (113,34) - (489,423) - length: 765 - iterations: 342223
Problem#9: (60,307) - (350,354) - length: 337 - iterations: 149614
Problem#12: (422,584) - (107,589) - length: 320 - iterations: 125153
Problem#6: (279,152) - (618,465) - length: 652 - iterations: 383761
Problem#11: (493,167) - (222,265) - length: 369 - iterations: 183582
Problem#18: (423,194) - (398,320) - length: 151 - iterations: 45554
Problem#19: (119,429) - (111,580) - length: 159 - iterations: 48948
Problem#17: (232,148) - (381,22) - length: 275 - iterations: 133651
Problem#20: (13,309) - (113,149) - length: 260 - iterations: 74511
Problem#15: (303,611) - (328,206) - length: 430 - iterations: 191063
Problem#24: (297,432) - (392,501) - length: 164 - iterations: 53931
Problem#21: (279,35) - (623,21) - length: 358 - iterations: 147055
Problem#16: (313,267) - (603,279) - length: 302 - iterations: 181243
Problem#27: (299,273) - (344,252) - length: 66 - iterations: 8754
Problem#14: (626,79) - (148,152) - length: 551 - iterations: 200691
Problem#13: (216,64) - (525,344) - length: 589 - iterations: 304957
Problem#26: (359,475) - (441,372) - length: 185 - iterations: 68236
Problem#10: (393,490) - (64,8) - length: 811 - iterations: 406964
Problem#29: (553,90) - (379,333) - length: 417 - iterations: 160783
Problem#25: (436,382) - (324,618) - length: 348 - iterations: 213150
Problem#22: (413,605) - (248,223) - length: 547 - iterations: 260947
Problem#23: (130,454) - (548,561) - length: 525 - iterations: 296899
Problem#28: (431,263) - (150,389) - length: 407 - iterations: 270185
```

Hér prentast niðurstöðurnar út 8(PPOOL_SIZE) í senn og innan þessara 8, í þeirri röð sem problemin voru leyst. Við sjáum að fyrstu línurnar eru eingöngu problem á bilinu 0-7, næst koma problem 8-15 o.s.frv.

Ef RUN_SETTINGS er ekki 1, 2 eða 3 eru prentuð út skilaboð þar sem notandinn er beðinn um að passa að 1, 2 eða 3 sé valið.