EGCI 213

Group Project 2 – Tour Assignment Simulation

The project can be done in a group of <= 4 students. Each group must do the assignment by themselves. **Everyone involved in cheating will get ZERO point**

1. This project has only 1 input file config.txt.
   - 1.1 Line 1 has 1 column = days of simulation
   - 1.2 Line 2 has 1 column = maximum daily arrival
   - 1.3 Line 3 has 1 column = number of tours (n), followed by n lines of tour data. In each line of tour data: col 1 = tour name, col 2 = daily capacity
   - 1.4 Line after tour data has 1 column = number of agencies (m), followed by m lines of agency data. In each line of agency data: col 1 = agency name, col 2 = tour name

```
config.txt
5
50
2
Tour_1   50
Tour_2   100
4
Agency_1   Tour_1
Agency_2   Tour_1
Agency_3   Tour_2
Agency_4   Tour_2
```

** Don't hard code these values. I may change some of them to check whether your calculation is correct

- Number of days, max arrival, capacity may be changed
- Number of tours (n) and agencies (m) may be changed, but n is always followed by n lines of tours and m is always followed by m lines of agencies. Tour names, agency names, and their mapping may also be changed
- There won't be any input error in this file. But the program must still handle the case of missing file – don't let it crash

2. Implement **class Tour** that represents an individual tour. Tour name and its daily capacity to receive customers are read from config.txt.

3. Implement **class AgencyThread** that represents an individual agency. Agency name and the tour it needs to put customers on are read from config.txt. Thread activities are done in loop. Each iteration of the loop represents 1 day. In each day
   - 3.1 Let 1 thread (AgencyThread or main thread) print day number
   - 3.2 Random new arrival, which must not exceed max daily arrival. Calculate total remaining customers = new arrivals + remaining customers from previous days. Print new arrivals and total remaining customers
   - 3.3 All AgencyThreads must wait for each other until all finish step 3.2, then proceed to step 3.4
   - 3.4 Put customers on the tour as the tour capacity allows. Maximum concurrency is expected, i.e. threads need to synchronize only if they update the same tour at the same time. But if they update different tours, they don't need to synchronize. Print number of customers successfully put on the tour

4. Implement main class with main method
   - 4.1 Read data from config.txt
   - 4.2 Create Tours and AgencyThreads, then start AgencyThreads. For flexibility, you're recommended to use ArrayList<Tour> and ArrayList<AgencyThread>.
   - 4.3 After all AgencyThreads completing all days of simulation, let main thread report total arrivals and total success of all AgencyThreads, sorted in decreasing order of total arrivals.

** Everything printed to the screen must be labelled by the name of thread who prints it. Don't hard code thread's name but use Thread.currentThread().getName()

5. Package and folder structure must be correct
   - 5.1 Your source files (.java) must be in folder Project2_XXX where XXX = ID of the group representative, assuming that this folder is under Maven's "src/main/java" structure. The first lines of all source files must be comments containing names & IDs of all members.
   - 5.2 Input files must be read from Project2_XXX. Don't use absolute path that is valid only on your PC.
   - 5.3 Add readme.txt containing names & IDs of all members in Project2_XXX.

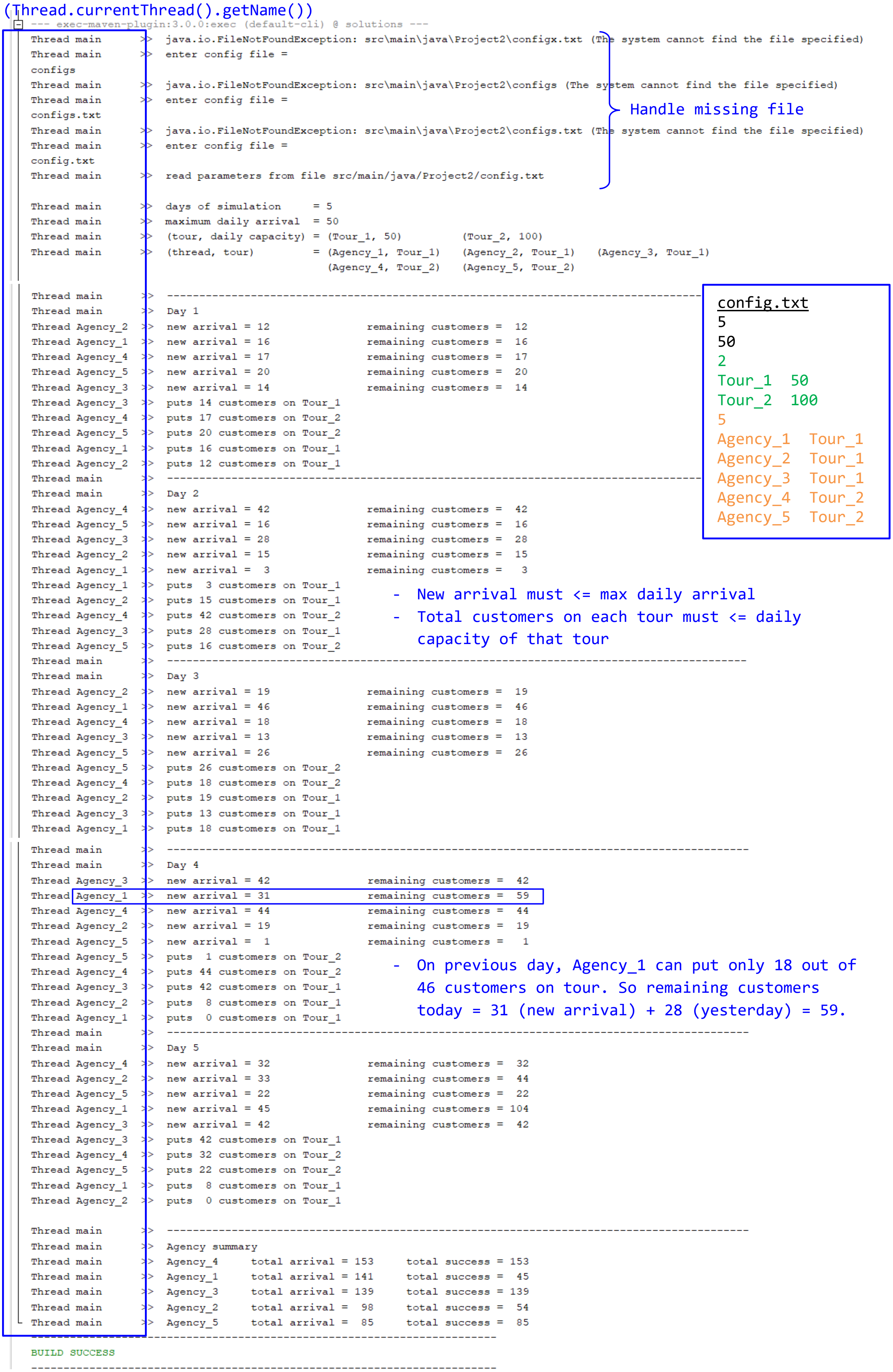
<u>Submission</u>

1. Group representative zips and submits Project2_XXX to Google classroom
2. Other members submit only readme.txt to Google classroom

<u>Grading</u>

3 points	correct customer update & tour capacity update in each day
1 point	looping from one day to another
1 point	correct summary report ** Different runs may give different results, but all the calculation must be correct
1 point	other requirements e.g. missing file handling, thread name, etc.
4 points	design & programming in proper OOP and multithreading style

All output lines are labeled with thread name
(Thread.currentThread().getName())



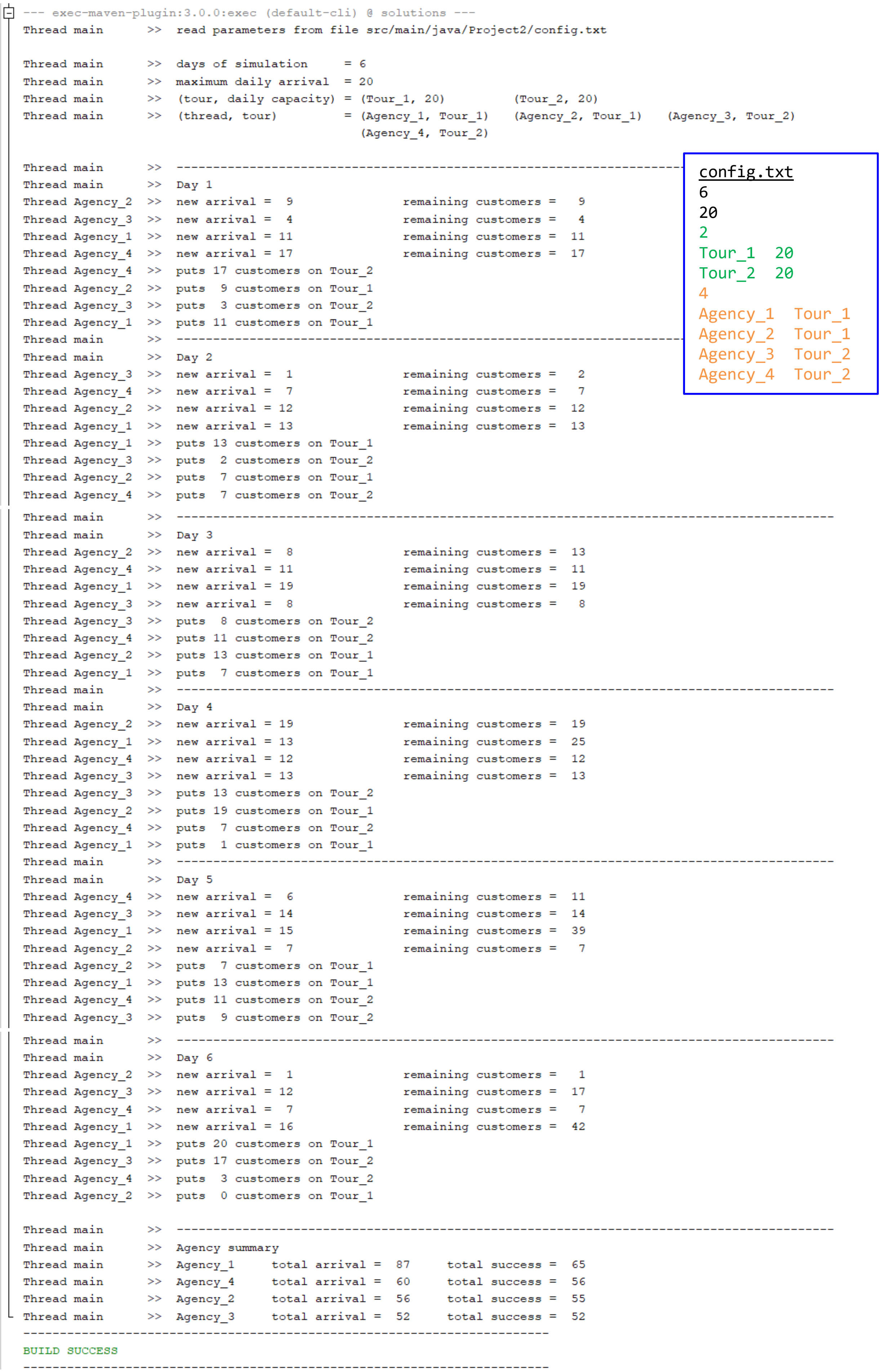
```
--- exec-maven-plugin:3.0.0:exec (default-cli) @ solutions ---
Thread main       >>  java.io.FileNotFoundException: src\main\java\Project2\configx.txt (The system cannot find the file specified)
Thread main       >>  enter config file =
configs
Thread main       >>  java.io.FileNotFoundException: src\main\java\Project2\configs (The system cannot find the file specified)
Thread main       >>  enter config file =
configs.txt
Thread main       >>  java.io.FileNotFoundException: src\main\java\Project2\configs.txt (The system cannot find the file specified)
Thread main       >>  enter config file =
config.txt
Thread main       >>  read parameters from file src/main/java/Project2/config.txt

Thread main       >>  days of simulation     = 5
Thread main       >>  maximum daily arrival  = 50
Thread main       >>  (tour, daily capacity) = (Tour_1, 50)         (Tour_2, 100)
Thread main       >>  (thread, tour)         = (Agency_1, Tour_1)   (Agency_2, Tour_1)   (Agency_3, Tour_1)
                                               (Agency_4, Tour_2)   (Agency_5, Tour_2)

Thread main       >>  ------------------------------------------------------------------------------------------
Thread main       >>  Day 1
Thread Agency_2   >>  new arrival = 12              remaining customers =  12
Thread Agency_1   >>  new arrival = 16              remaining customers =  16
Thread Agency_4   >>  new arrival = 17              remaining customers =  17
Thread Agency_5   >>  new arrival = 20              remaining customers =  20
Thread Agency_3   >>  new arrival = 14              remaining customers =  14
Thread Agency_3   >>  puts 14 customers on Tour_1
Thread Agency_4   >>  puts 17 customers on Tour_2
Thread Agency_5   >>  puts 20 customers on Tour_2
Thread Agency_1   >>  puts 16 customers on Tour_1
Thread Agency_2   >>  puts 12 customers on Tour_1
Thread main       >>  ------------------------------------------------------------------------------------------
Thread main       >>  Day 2
Thread Agency_4   >>  new arrival = 42              remaining customers =  42
Thread Agency_5   >>  new arrival = 16              remaining customers =  16
Thread Agency_3   >>  new arrival = 28              remaining customers =  28
Thread Agency_2   >>  new arrival = 15              remaining customers =  15
Thread Agency_1   >>  new arrival =  3              remaining customers =   3
Thread Agency_1   >>  puts  3 customers on Tour_1
Thread Agency_2   >>  puts 15 customers on Tour_1
Thread Agency_4   >>  puts 42 customers on Tour_2
Thread Agency_3   >>  puts 28 customers on Tour_1
Thread Agency_5   >>  puts 16 customers on Tour_2
Thread main       >>  ------------------------------------------------------------------------------------------
Thread main       >>  Day 3
Thread Agency_2   >>  new arrival = 19              remaining customers =  19
Thread Agency_1   >>  new arrival = 46              remaining customers =  46
Thread Agency_4   >>  new arrival = 18              remaining customers =  18
Thread Agency_3   >>  new arrival = 13              remaining customers =  13
Thread Agency_5   >>  new arrival = 26              remaining customers =  26
Thread Agency_5   >>  puts 26 customers on Tour_2
Thread Agency_4   >>  puts 18 customers on Tour_2
Thread Agency_2   >>  puts 19 customers on Tour_1
Thread Agency_3   >>  puts 13 customers on Tour_1
Thread Agency_1   >>  puts 18 customers on Tour_1

Thread main       >>  ------------------------------------------------------------------------------------------
Thread main       >>  Day 4
Thread Agency_3   >>  new arrival = 42              remaining customers =  42
Thread Agency_1   >>  new arrival = 31              remaining customers =  59
Thread Agency_4   >>  new arrival = 44              remaining customers =  44
Thread Agency_2   >>  new arrival = 19              remaining customers =  19
Thread Agency_5   >>  new arrival =  1              remaining customers =   1
Thread Agency_5   >>  puts  1 customers on Tour_2
Thread Agency_4   >>  puts 44 customers on Tour_2
Thread Agency_3   >>  puts 42 customers on Tour_1
Thread Agency_2   >>  puts  8 customers on Tour_1
Thread Agency_1   >>  puts  0 customers on Tour_1
Thread main       >>  ------------------------------------------------------------------------------------------
Thread main       >>  Day 5
Thread Agency_4   >>  new arrival = 32              remaining customers =  32
Thread Agency_2   >>  new arrival = 33              remaining customers =  44
Thread Agency_5   >>  new arrival = 22              remaining customers =  22
Thread Agency_1   >>  new arrival = 45              remaining customers = 104
Thread Agency_3   >>  new arrival = 42              remaining customers =  42
Thread Agency_3   >>  puts 42 customers on Tour_1
Thread Agency_4   >>  puts 32 customers on Tour_2
Thread Agency_5   >>  puts 22 customers on Tour_2
Thread Agency_1   >>  puts  8 customers on Tour_1
Thread Agency_2   >>  puts  0 customers on Tour_1

Thread main       >>  ------------------------------------------------------------------------------------------
Thread main       >>  Agency summary
Thread main       >>  Agency_4     total arrival = 153     total success = 153
Thread main       >>  Agency_1     total arrival = 141     total success =  45
Thread main       >>  Agency_3     total arrival = 139     total success = 139
Thread main       >>  Agency_2     total arrival =  98     total success =  54
Thread main       >>  Agency_5     total arrival =  85     total success =  85
------------------------------------------------------------------------

BUILD SUCCESS
------------------------------------------------------------------------
```

Handle missing file

```
config.txt
5
50
2
Tour_1  50
Tour_2  100
5
Agency_1  Tour_1
Agency_2  Tour_1
Agency_3  Tour_1
Agency_4  Tour_2
Agency_5  Tour_2
```

- New arrival must <= max daily arrival
- Total customers on each tour must <= daily capacity of that tour

- On previous day, Agency_1 can put only 18 out of 46 customers on tour. So remaining customers today = 31 (new arrival) + 28 (yesterday) = 59.



```
--- exec-maven-plugin:3.0.0:exec (default-cli) @ solutions ---
Thread main      >>  read parameters from file src/main/java/Project2/config.txt

Thread main      >>  days of simulation     = 6
Thread main      >>  maximum daily arrival  = 20
Thread main      >>  (tour, daily capacity) = (Tour_1, 20)         (Tour_2, 20)
Thread main      >>  (thread, tour)         = (Agency_1, Tour_1)   (Agency_2, Tour_1)   (Agency_3, Tour_2)
                                              (Agency_4, Tour_2)

Thread main      >>  -------------------------------------------------------------------------------------
Thread main      >>  Day 1
Thread Agency_2  >>  new arrival =  9             remaining customers =   9
Thread Agency_3  >>  new arrival =  4             remaining customers =   4
Thread Agency_1  >>  new arrival = 11             remaining customers =  11
Thread Agency_4  >>  new arrival = 17             remaining customers =  17
Thread Agency_4  >>  puts 17 customers on Tour_2
Thread Agency_2  >>  puts  9 customers on Tour_1
Thread Agency_3  >>  puts  3 customers on Tour_2
Thread Agency_1  >>  puts 11 customers on Tour_1
Thread main      >>  -------------------------------------------------------------------------------------
Thread main      >>  Day 2
Thread Agency_3  >>  new arrival =  1             remaining customers =   2
Thread Agency_4  >>  new arrival =  7             remaining customers =   7
Thread Agency_2  >>  new arrival = 12             remaining customers =  12
Thread Agency_1  >>  new arrival = 13             remaining customers =  13
Thread Agency_1  >>  puts 13 customers on Tour_1
Thread Agency_3  >>  puts  2 customers on Tour_2
Thread Agency_2  >>  puts  7 customers on Tour_1
Thread Agency_4  >>  puts  7 customers on Tour_2
Thread main      >>  -------------------------------------------------------------------------------------
Thread main      >>  Day 3
Thread Agency_2  >>  new arrival =  8             remaining customers =  13
Thread Agency_4  >>  new arrival = 11             remaining customers =  11
Thread Agency_1  >>  new arrival = 19             remaining customers =  19
Thread Agency_3  >>  new arrival =  8             remaining customers =   8
Thread Agency_3  >>  puts  8 customers on Tour_2
Thread Agency_4  >>  puts 11 customers on Tour_2
Thread Agency_2  >>  puts 13 customers on Tour_1
Thread Agency_1  >>  puts  7 customers on Tour_1
Thread main      >>  -------------------------------------------------------------------------------------
Thread main      >>  Day 4
Thread Agency_2  >>  new arrival = 19             remaining customers =  19
Thread Agency_1  >>  new arrival = 13             remaining customers =  25
Thread Agency_4  >>  new arrival = 12             remaining customers =  12
Thread Agency_3  >>  new arrival = 13             remaining customers =  13
Thread Agency_3  >>  puts 13 customers on Tour_2
Thread Agency_2  >>  puts 19 customers on Tour_1
Thread Agency_4  >>  puts  7 customers on Tour_2
Thread Agency_1  >>  puts  1 customers on Tour_1
Thread main      >>  -------------------------------------------------------------------------------------
Thread main      >>  Day 5
Thread Agency_4  >>  new arrival =  6             remaining customers =  11
Thread Agency_3  >>  new arrival = 14             remaining customers =  14
Thread Agency_1  >>  new arrival = 15             remaining customers =  39
Thread Agency_2  >>  new arrival =  7             remaining customers =   7
Thread Agency_2  >>  puts  7 customers on Tour_1
Thread Agency_1  >>  puts 13 customers on Tour_1
Thread Agency_4  >>  puts 11 customers on Tour_2
Thread Agency_3  >>  puts  9 customers on Tour_2
Thread main      >>  -------------------------------------------------------------------------------------
Thread main      >>  Day 6
Thread Agency_2  >>  new arrival =  1             remaining customers =   1
Thread Agency_3  >>  new arrival = 12             remaining customers =  17
Thread Agency_4  >>  new arrival =  7             remaining customers =   7
Thread Agency_1  >>  new arrival = 16             remaining customers =  42
Thread Agency_1  >>  puts 20 customers on Tour_1
Thread Agency_3  >>  puts 17 customers on Tour_2
Thread Agency_4  >>  puts  3 customers on Tour_2
Thread Agency_2  >>  puts  0 customers on Tour_1

Thread main      >>  -------------------------------------------------------------------------------------
Thread main      >>  Agency summary
Thread main      >>  Agency_1     total arrival =  87     total success =  65
Thread main      >>  Agency_4     total arrival =  60     total success =  56
Thread main      >>  Agency_2     total arrival =  56     total success =  55
Thread main      >>  Agency_3     total arrival =  52     total success =  52
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
```

config.txt

```
6
20
2
Tour_1   20
Tour_2   20
4
Agency_1   Tour_1
Agency_2   Tour_1
Agency_3   Tour_2
Agency_4   Tour_2
```