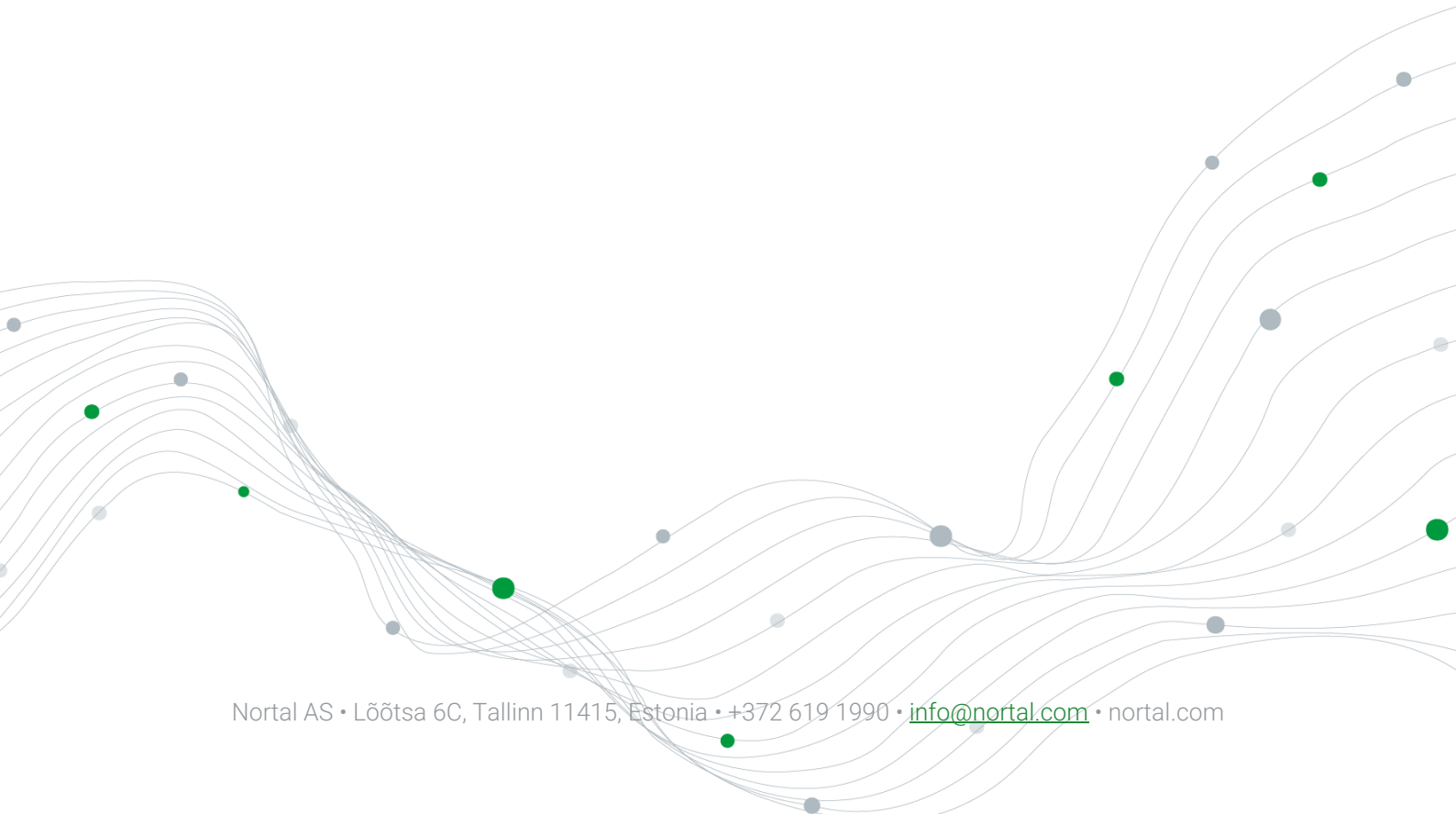
Nortal

SUMMER UNIVERSITY 2022

JAVA & TEST AUTOMATION EXERCISE



Nortal AS • Lõõtsa 6C, Tallinn 11415, Estonia • +372 619 1990 • info@nortal.com • nortal.com

Nortal

Summer University 2022

And the challenge begins.

There is one task in this test assignment. The description for it is given below. Solve as much as you can - a partial solution is better than no solution. In addition to making your code work, make it clean as well.

Good luck!

Submitting Assignment

Once you have completed your development, run './gradlew createSubmissionZip' in the root directory, where this README.pdf resides. Your results will be packed into assignment_submit.zip under build/ directory. This is the file that you need to upload.

Tips

- If something doesn't work – google it.
- Minimal Gradle instruction discrepancies may occur. It all depends on your OS (Operating System) and even the console (Command line) you use.
- We strongly recommend using IntelliJ IDEA development environment for your test assignment and work in general.

Nortal

Hummingbird Travel

Hummingbirds are tiny birds that can travel about 2200 kilometers without having to stop. It's about 29 million times the length of their body. Meanwhile, the marathon-trained developer has a ratio of around 25 thousand.

You're a curious traveler who found a strange species of these little birds. When gathered, they seem to form a rectangular shape. It moves in a strange, but beautiful and organized way. You notice the patterns and understand that it's a game! As a passionate developer, you start to think about the algorithm that is happening here. You decide to calculate how long this 'dance' will last.

Setup

On Windows, execute './gradlew build'. On Linux execute 'gradlew build'. You can also use your own locally installed Gradle, but it might result in mismatching the Gradle project root folder. Alternatively, skip these previous steps by installing a smart IDE, like IntelliJ IDEA, which automates it.

Assignment

Read the game map from the TXT file and count the number of game iterations until the game is blocked.

Game Rules

- The map consists of squares stored in a TXT file (see src/main/resources/ hummingbird_map.txt).
- The map is always rectangular-shaped of size M x N.
- 0 < M <= 100,000
- 0 < N <= 100,000
- Only one bird occupies one square of the rectangular grid at one point in time.
- There are empty squares (.) . Existing squares that are not occupied by a hummingbird.
- There are gaps in the map (x). Non-existent squares/holes in the map.
- There are two types of hummingbirds. Ones moving right (>) and ones moving down (v).
- Each bird moves in straight lines (one direction) only.
- When crossing the border of the playground, the hummingbird gets straight to the other end of the current position. (You can hardly even notice it; They are so quick!).
- Right-facing birds move first. They first evaluate if the next square is empty and then move simultaneously,
- Down-facing birds move second. They first evaluate if the next square is empty and then move simultaneously,
- If the birds never stop moving, print the number of iterations, when you see the first repetitive composition. Previously seen arrangement of birds on the ‘game board‘.

Other Requirements

- The application should not be communicating with the console
- You can’t make any changes in the HummingbirdGame.java file

Example Game 1

```
Start:  .x..>.
Step 1: .x...>
Step 2: >x....
Step 3: .x>...
Step 4: .x.>..
...
```

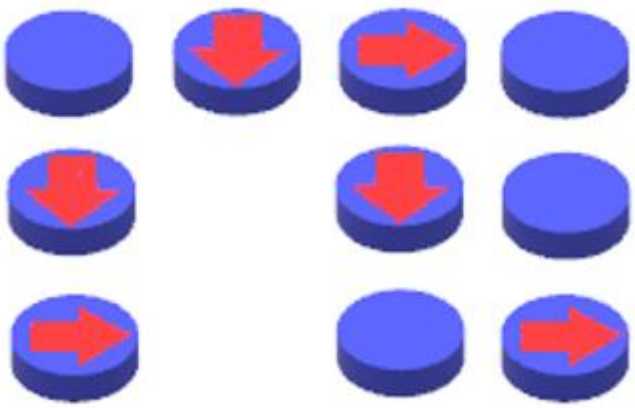
The expected output is int 5.

Example Game 2

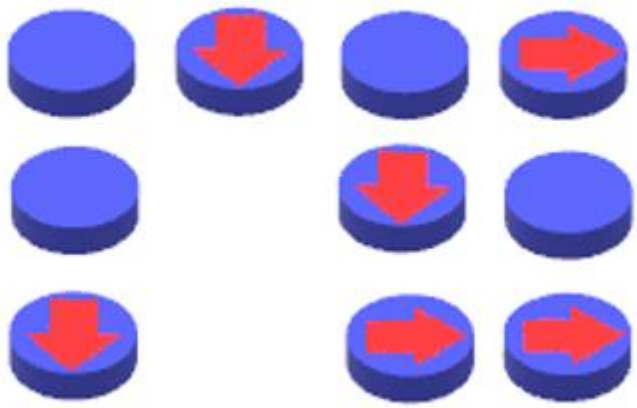
Imagine the following TXT file (map):

```
.v>.
vxv.
>x.>
```

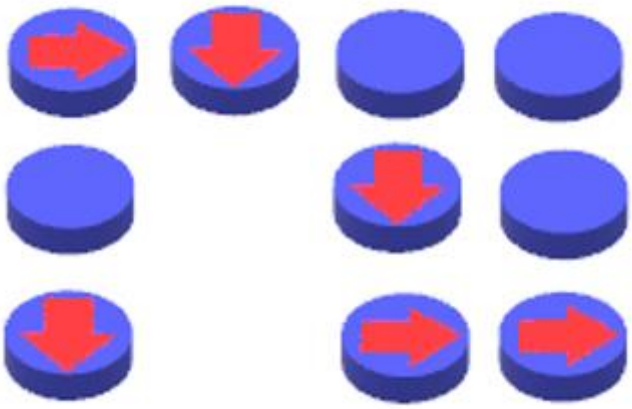
It can be visualized as (initial state):



In the first round, right-facing birds will move one step to the right if the next platform is empty. Then, down-facing birds will also move one step down if it's not against rules (State after one round).



Same happens on round two (state after two rounds, final state).



Neither right nor down facing birds has eligible moves, so the game is over after two rounds.

The expected output is int 2.

Nortal

Additional Important Instructions

TODOs have been left in the HummingbirdGameImpl.java file. Implementation is up to you. Just remember, you **can't make any changes in the HummingbirdGame.java** interface. This is the only restriction you have.

Do you have questions?

As the forum is open (https://nortal.com/careers/summeruniversity-lt-22/java-task-discussion-lt-2022/), read the previous answers and if you didn't find the answer, ask your own. Our specialists keep an eye on the forum every day and answer your questions as soon as possible.