Knights and chessboards

Dear applicant,

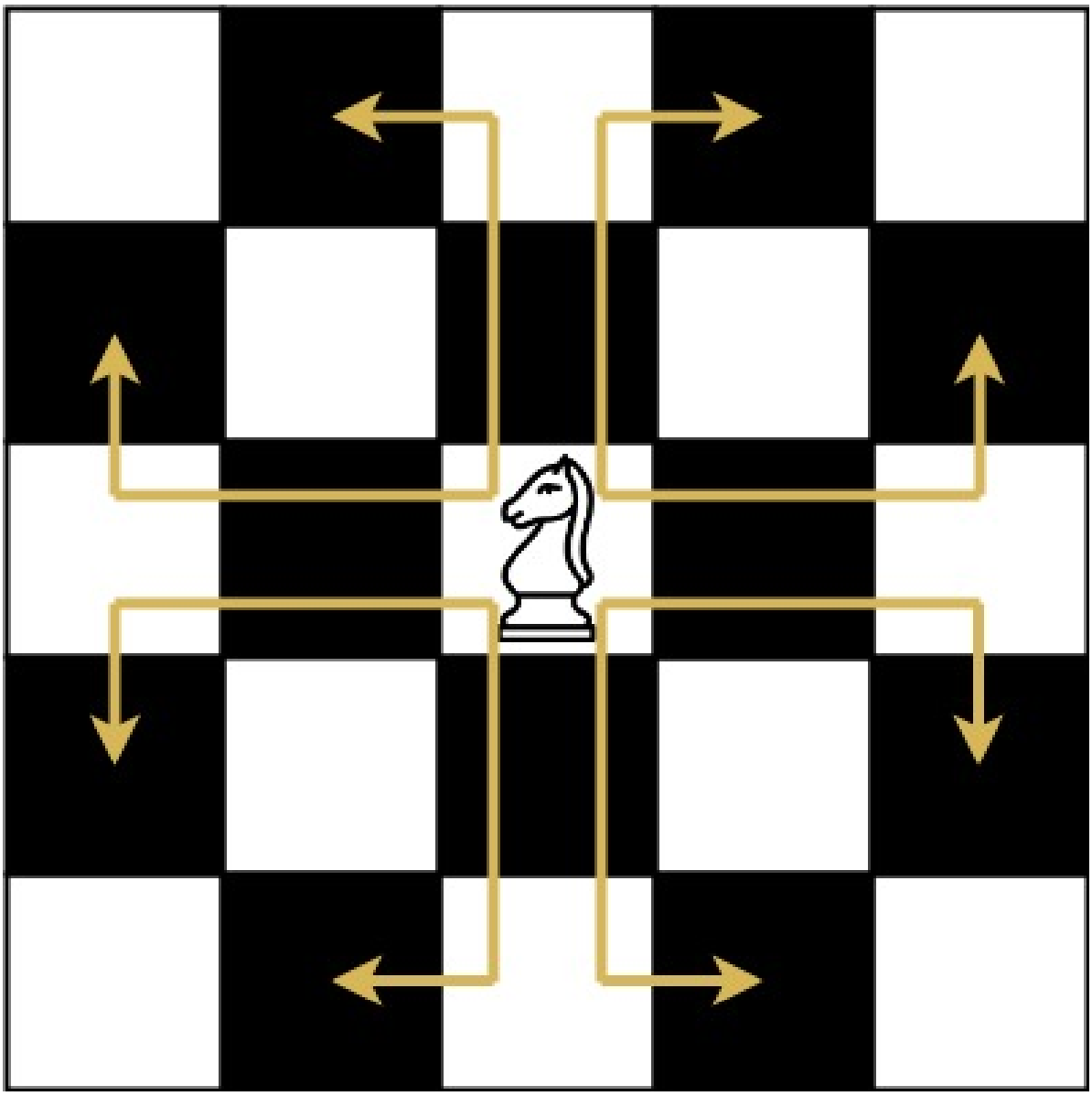
In order to get a better picture of your programming skills, we would like to ask you to solve this small problem by Monday. For this you have to use a common object oriented programming language, e.g. Java, C, C#, C++, Scala. You are free to send us your code as a reply to this e-mail or upload it to your Github and send us the link. In either case, please include a short instruction on how to execute your solution.

If you have any questions please contact [redacted]

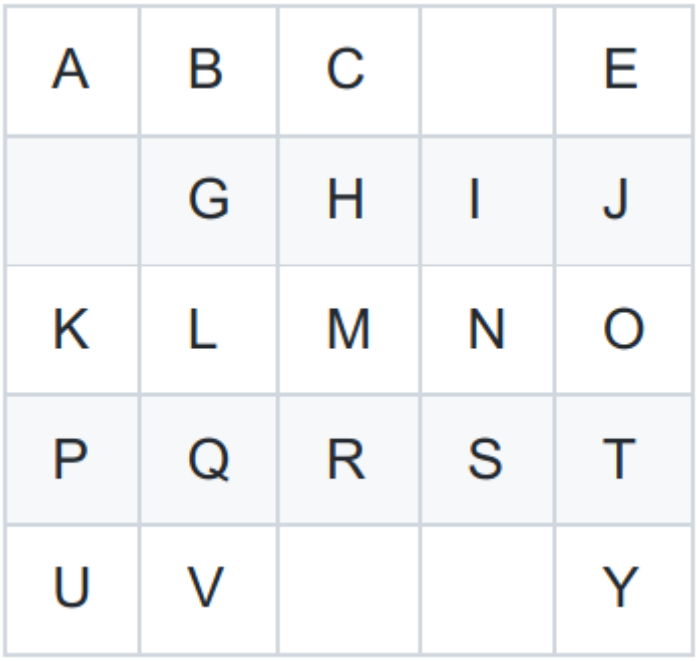
The task

The chess knight has a **unique movement**, it may move two squares vertically and one square horizontally, or two squares horizontally and one square vertically (with both forming the shape of an **L**). The possible movements of chess knight are shown in this diagram:

A chess knight can move as indicated in the chess diagram below:



Given the following miniature chessboard:



Given an integer `n`, return how many distinct strings can be produced by a knight moving from square to square for `n` times . The following requirements must be fulfilled:

1. Only cells containing a character are considered **valid**.
2. You are allowed to place the knight **valid cell** initially and then you should perform `n - 1` jumps to build the string. The character of the cell you placed the knight in, is also part of the resulting string.
   Example 1:

```
Input: n = 1
Output: 21
Explanation: We need to produce a string of length 1, so placing the knight
over any valid cell is sufficient.
```

3. A **valid** knight jump is not allowed to start or end on an **non valid cell**.
4. All jumps should be **valid** knight jumps.
5. The characters/cells that the knight jumps over during a jump are not part of the string produced.
6. The resulting string cannot contain more than two vowels.