

# Bonus assignment 14: Solving the Ice Sliding Puzzle – the penguins version

Course ‘Imperative Programming’ (IPC031)

## 1 Background

In this version of the Ice Sliding puzzle the rocks are replaced by penguins (there might also be zero penguins).<sup>1</sup> Again, there is exactly one rescue location that the flamingo needs to reach. The flamingo and the penguins can move into the directions north, east, south, and west. As soon as they start moving, they keep sliding on the ice until they either slide off the ice or bump into a penguin or flamingo. Moreover, only the flamingo stands still immediately as it reaches the rescue location. When bumping into another animal, the sliding animal stands still immediately before the animal.<sup>2</sup> Penguins that slide off the ice floe start swimming and do not return, but the flamingo drowns if it slides off the ice. At any point in time only one animal can perform a move. The puzzle is finished if the flamingo has reached the rescue location.

## 2 Assignment

On Brightspace, you find the file “`assignment-14-bonus-files.zip`”. It contains a number of text files with names “`challenge.m.steps.txt`”, where `m` is the number of the challenge and `n` is the least number of steps required to solve the challenge. In these text files a puzzle starting configuration is described by a matrix of cells that have the following representation:

- an empty ice cell by ‘.’
- a penguin by ‘p’, and if it is on the rescue location by ‘P’
- the flamingo by ‘f’, and if it is on the rescue location by ‘F’
- the empty rescue location by ‘x’

Adopt your solution for the mandatory assignment to work with this puzzle variant. In case you completed the prior bonus assignment, you should already have all the puzzle logic needed, which you can copy over. You will probably only need to make some minor changes to the search algorithms. Use the provided test cases to verify that your implementation works correctly.

## Important

Some challenges, such as 4 and 13—especially with a depth first search—may take a long time to complete, even with a decently optimized solution. For development purposes you may want to disable those test cases by commenting them out, until you are confident you have a working solution. Our internal reference solution manages to solve all test cases in roughly 2 minutes.<sup>3</sup> Perhaps you want an extra challenge to show off all that you have learned, in which case: try beating our time!

## 3 Products

As product-to-deliver upload to Brightspace “`main.cpp`” that you have created with your solution and “`main_test.cpp`”.

## Deadline

**Bonus assignment:** Monday December 25, 2023, 15:30

<sup>1</sup>We were unable to trace back the origins of this particular ice sliding puzzle version, but it seems that they were invented by Nick de Ruyck.

<sup>2</sup>We ignore the conservation of momentum, meaning the animal being bumped into remains stationary.

<sup>3</sup>Running all tests sequentially on a single core, so no multi-threaded magic.