



EXERCISES — Lakes of Finland

version #



IT IS MY JOB TO MAKE SURE YOU DO YOURS.

Copyright

This document is for internal use at EPITA ([website](#)) only.

Copyright © 2021-2022 Assistants <assistants@tickets.assistants.epita.fr>

The use of this document must abide by the following rules:

- ▷ You downloaded it from the assistants' intranet.*
- ▷ This document is strictly personal and must **not** be passed onto someone else.
- ▷ Non-compliance with these rules can lead to severe sanctions.

Contents

1	Lakes of Finland	3
1.1	Goal	3
1.2	Example	4

*<https://intra.assistants.epita.fr>

1 Lakes of Finland

Files to submit:

- lakes/lakes.c

Provided files:

- lakes/lakes-example.c
- lakes/lake.in

Authorized functions: You are only allowed to use the following functions:

- malloc(3)
- free(3)

Authorized headers: You are only allowed to use the functions defined in the following headers:

- assert.h
- err.h
- errno.h
- stddef.h

1.1 Goal

A geographical map is represented as a 2D array of characters. Each character can be either '#' (ground) or '.' (water). The purpose of this exercise is to count the number of lakes on the map.

A lake is a water area made of connected water delimited by ground. Two water cells are said to be connected if they are vertically or horizontally adjacent. That means that two water cell diagonally adjacent are not connected:

```
##
.. connected

#.
#. connected

#.
.# not connected
```

In order to make the problem simpler, there will not be any map with water on the border. The minimum number of lakes is 0, which means there can be a map without any lake.

You have to implement the following function:

```
int lakes(char **map, int width, int height);
```

- map represents the map
- width and height are the map dimensions

You will not have to handle error cases.

1.2 Example

The given `lakes-example.c` results to:

[illegible]

It is my job to make sure you do yours.