

# Mini project: "Enemies" game

## Make a single-player game.

---

### Description

The player controls the main character and the purpose is to avoid enemies and survive until all enemies are dead.

The main character can move in 4 directions or jump to a random position on the board.

Enemies chase the main character all the time by moving in 8 directions and they are eliminated if they collide with an obstacle or another enemy.

---

### Requirements

The game should have:

- A game board, consisting of fields.
- Obstacles
- Enemies
- A main character, controlled by the human player

Every time the player moves, all enemies will move one field in the direction of the player/main character.

The game ends when the main character is eliminated (by colliding with an enemy or an obstacle) or when all enemies are eliminated.

---

### Gameplay

When the game starts, a number of enemies and obstacles are placed randomly on the board.

One version (the recommended) could be turn-based where time stops after each move.

Another version (advanced) could be real-time.

A third version could be turn-based with a timer, where the user has to make a move before x seconds, or the next turn will start.

---

### Strategy

The user plays the game by deciding which move the main character should make (up, down, left, right, jump). Enemies are always moving in the direction of the main character, and are not aware of the obstacles, so a strategy for eliminating enemies is to “lure” the enemies to walk into obstacles or other enemies.

---

### Suggestions

One suggestion for improving gameplay could be to restrict the number of jumps available. It could e.g. be a fixed number of jumps available or a “cool-down” time...

---

## Example layout

Use specific characters to mark specific objects, such as:

# an obstacle  
X an enemy  
A the player

Assuming a gameboard size 10 x 10 fields, a snapshot of a running game could be shown as:

```
*****
*      #      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*****
```

If the player now moves one character up:

```
*****
*      #      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*****
```

the enemies move one step towards the player:

```
*****
*      #      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*      *      *
*****
```

Resulting in two enemies running into obstacles and dying and two enemies running into each other resulting in both to die.

---

## Evaluation Criteria

Your program will be evaluated on the basis of the following elements;

- Functionality. The Application should work as expected.
- Reflection of the course. You are expected to use elements from the course in your solution.
- Well organized and commented code. a good simple architecture with clear logical breakdown of the code in the methods and classes that helps to a better understanding of the code.
- Error Handling. Try to make sure that the application is solid and can handle incorrect input and or other possible errors.
- Project Description. In order to make sure your ideas and code isn't misunderstood add a short PDF document (max 2 pages) describing the reasoning behind your solution.

---

## About the exam

The exam will be a 15 minute individual oral exam. The exam will mainly be about the assignments (especially the mini project) but questions can be related to the whole course curriculum.

---

## Plagiarism

Notice that it is OK to find inspiration on the Internet, but DO NOT copy paste! If you choose to use parts of other people's code make sure you a) give them credit for it and b) understand their code fully as you will be examined in all the code as had you written it yourself.

If you find code somewhere and want to use it as a part of your solution you should add a comment and write exactly which parts of the code you have copied, and where it has been copied from (give the URL).

Take ownership of the code - You must be able to explain any part of the source code.

---

## Working together in groups

You can work alone, in pairs, or groups of up to 3 persons. Clearly state the names of all group members on each page of the documentation.

---

## Deadline

Your solution (.java files in a zip archive) and description (as a PDF document) must be uploaded to [eksamen.ruc.dk](https://eksamen.ruc.dk) no later than Friday November 19th at 10:00 cest.