

[yoavg](#) / [ioop2017](#)

# Assignment 7

Yoav Goldberg edited this page on 17 Jun 2017 · 5 revisions

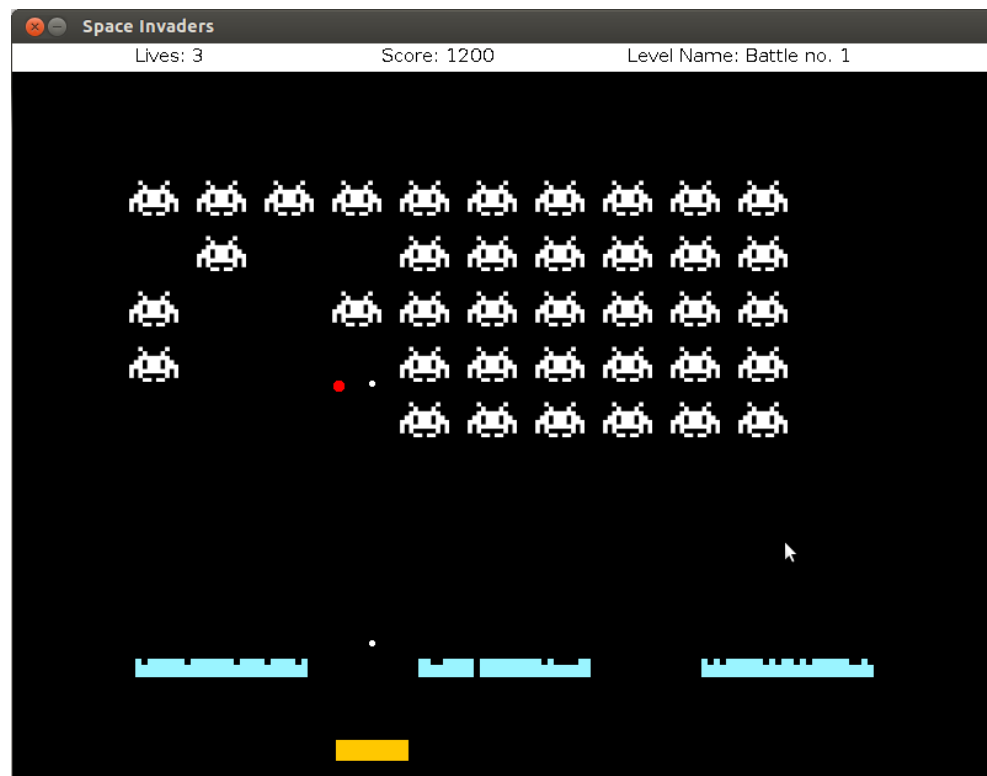
Deadline: July 2, 2017

## Code Reuse - Space Invaders

### Introduction

This assignment builds on top of the code that you developed for the Arkanoid game. However, rather than adding features to the nice game we have, we will implement a brand new game.

We will be implementing a version of the classic 80s game [Space Invaders](#).



Unlike the previous assignments in which we directed you in every step of the way, in this assignment you are on your own in terms of design. We only provide the requirements, and a few hints below.

### Requirements

You are expected to implement a game with the same functionality as the game in this [example jar](#) file.

To run:

```
java -jar space-invaders.jar
```

Specifically:

- Screen size is 800 x 600
- There is a player controlled space-ship that moves left and right at the bottom of the screen.
- Above the space-ship, there are "shields" that are slowly destroyed when they are shot.

► Pages 21

- [piazza](#)
- [Assignments](#)

Clone this wiki locally

<https://github.com/yoavg/io>



Clone in Desktop

- Above the shields, there are aliens.
  - The aliens move in a formation.
  - When the formation hits the left or right side of the screen, the aliens in the formation (a) go down; (b) change direction; and (c) increase their speed by 10%. The formation is considered to hit the side of the screen when the left-most (or right-most) alien in the formation hits the side of the screen.
  - When the lowest alien in the formation reaches the height of the shields (or where the shields were, if all shields are destroyed), the player loses a life.
- Aliens can shoot bullets.
  - When the bullet hits the shield, it puts a small hole in it. When the bullet hits the spaceship, the player loses a life.
  - Every 0.5 seconds a random column of aliens is chosen to shoot, the lowest alien on that column will release a shot.
  - If an alien bullet hits another alien it should disappear without killing that alien.
- When the player loses a life, the aliens reset their position back to the top, returning to their original speed (shields remain unchanged).
- The player can also shoot bullets (pressing or holding the 'space' button). When the bullets hit the shields, it puts a small hole in them. When the bullet hits an alien, the alien dies.
  - **The player can only shoot a bullet every 0.35 seconds**
- When all the aliens die, the player wins.
- The player starts the game with 3 lives.
- Scoring:
  - Each alien is worth 100 points.
- Levels:
  - Each level is the same except for different initial speed of the aliens. Each level the speed will increase to make the game more challenging.
  - There is an infinite amount of levels so the game will only end by the player losing.
- Other:
  - We expect there to be a menu similar to the last Arkanoid version we implemented (the only difference is that there will be no choice of level sets).
  - Highscore tracking should also work just like the last Arkanoid version we implemented.

## Notes

---

- You can re-use many of the classes you wrote for the Arkanoid game, without changing their code.
- In particular, you can reuse the blocks, balls, sprites and notification mechanisms. Think of what game elements can be blocks. Remember that blocks can be very small.
- Aliens move in a group. Consider creating a new class (or more than one) to coordinate the group's movement.
- The aliens and their formation:
  - Each alien's size is 40 by 30 pixels.
  - There are 10 columns and 5 rows of aliens.
  - There is a 10 pixel gap between the rows and the columns.
  - You can use [this image](#) for the aliens or create one yourself.

## Code Description

---

You will need to submit a `pdf` file called `README.pdf` describing your solution. In particular, it should include:

- The list of interfaces and classes in your code, and for each one a description of its purpose and responsibilities.
- A brief description of how you implemented the following: (a) the Aliens formation; (b) the shields; (c) shots by aliens; (d) shots by player.

## What to submit:

---

You need to submit a file called `ass7.zip` containing everything needed to compile and run your game (source code, resources).

The file should also contain the `README.pdf` file, in the main folder.

Your `compile` target should compile all the classes.

Your single `run` target should run the game (without parameters).

Your `jar` target should produce a file called `space-invaders.jar`. We should be able to run your `.jar` file using the command:

```
java -jar space-invaders.jar
```

The jar file should be created in the same directory as the makefile. To be clear, we should be able to use the following commands to run your code:

```
unzip ass7.zip
make compile
make jar
java -jar space-invaders.jar
```

Like in the previous assignment, it is OK to have the `hashCode` checkstyle error.