1 Overview
For the fourth assignment you will implement a simple game in JavaScript. The game consists of a single web page. You will be provided a sample HTML file together with a style sheet and images it includes and you will have to write the JavaScript file implementing the logic of the game. Your solution will only be tested in Firefox.

2 Game description
The name of the game is “Catch the prof” and the goal of the player is to click on the images of all professors as fast as possible. To make the game more challenging these images will be moving continuously.

After the user clicks on the picture of one of the profs, it should move to the “Jail” area. While still in the “Arena” area, the images should be moving in a straight line until they
hit the border, at which time they would bounce in a new direction. The images should never move outside the arena. Images may overlap while in the arena. In the jail, they should not overlap.

You should select for each image random starting locations within the arena. When an image collides with a border of the arena it should start moving in a random direction without crossing the border. They should also occasionally move along the border. The speed of movement should also be random and non-zero.

The radio buttons in the speed group should control the range of random speeds: if “fast” is selected then after some of the collisions the images will move fast, if “slow” is selected, the images should move slowly. You can wait until the next collision to change the speed of an image in response to the user selecting another radio button. The “Start” button should move all images to random positions in the arena and start their movement. The “Stop” button should stop the movement of all images. The user may push these buttons repeatedly. When the page is loaded, it will call the `initializeTheGame()` function which should also place all images to random positions in the arena.

3 Hints and useful bits of information

To implement random linear movements for an image use two values $\Delta x$ and $\Delta y$ initialized to random integer values between $-\text{speed}$ and $\text{speed}$ where $\text{speed}$ is the JavaScript variable controlled by the radio buttons. Whenever the timer fires you can increment the images $x$ position by $\Delta x$ and its $y$ position by $\Delta y$ by controlling some specific style attributes. When $\Delta x$ is 0 the image will be moving vertically (possibly along one of the two vertical borders and when $\Delta y$ is 0 it will be moving horizontally (possibly along the top or bottom border).

You can use `Math.random()` to generate a random number between 0 and 1 and `Math.round()` to round a floating point number to the nearest integer. In Firefox you can use the `offsetTop` and `offsetLeft` properties of the objects representing the divs for the arena and the jail to read the coordinates of their upper left corner.

Use large timeout when debugging at first. Ctrl-alt-del should start your task manager which can use to kill your browser if it becomes unresponsive due to an infinite loop or some other JavaScript problem.

The first argument of `setTimeout` is a string that will be interpreted when the timeout fires as a JavaScript program that needs to be run. Note that if you have a variable $x$ with value 5 at the time when you call `setTimeout`, and you pass “someFunction($x$)” as its first, when `someFunction` is called it will not receive the argument 5 but the value of the global variable $x$ (if such a global variable exists).

All images are 75 by 84 pixels.
5 Grading criteria
The grading criteria for this assignment will include: the extent to which all the required features are covered and correctness. Correctness includes the correct handling of the case when an image gets to a corner of the arena and correct functioning of the stop button after the play button is pressed multiple times. It is not allowed for an image in the arena to react to a collision with a border by staying in one place.

6 Submission
Please email as attachment to a single message to the TA the JavaScript file implementing the game. If you modified the web page and/or the style sheet submit an archive with all files.