INFO I211 - Information Infrastructure II

Spring 2014

Assignment 2

A game in Javascript

(to be completed in your I211 Student Team)
Total: 100 points (and extra tasks for bonus 20+20 points, see below)

Due Date:

Friday, April 25 2014, 11:59PM, on IU OnCourse.

How to work on your Assignment 2:

The Assignment 2 is to be completed in your I211 Student Team, where each team member is to pick *one* of three/four tasks to implement the "*runner and chaser*" game, as presented at lecture time. The tasks are:

1. Task 1 = implement the "*runner*" character (if there was a team member who performed the "*document the process*" task in I211 Assignment 1, that person is assigned Task 1 for I211 Assignment 2).
2. Task 2 = implement the "*chaser*" character.
3. Task 3 = implement the playing field for the game.
4. Task 4 = write the comments for your code.
    - If there are four (4) students in your I211 Student Team, the task for the fourth team member is to provide comments for *all* the Javascript code and HTML tags for all included files (including the parts that are already provided to you in the "I211-assignment-2-starting-text.html" file contents).
    - If there are three (3) students in your I211 Student Team, every team member is to provide comments for the Javascript code and HTML tags for *their own task (1 to 3)* only.

Assignment Tasks:

The goal for Assignment 2 is to complete the "*runner and chaser*" game, as presented at I211 lectures 22 and 23. You have worked on parts of the game implementation, as lecture *group work* and lab *group tasks*. Please use your solutions for those tasks when you work on completing I211 Assignment 2.

Task 1

The first task is to complete the game logic for moving the "*runner*" character in the "*runner and chaser*" game, as presented at I211 lecture 22 pages 1 to 5.

1. The "*runner*" character in the game is represented by the "*at sign*" letter symbol "@". At any given time in the game, the "*runner*" has to be shown in only one location in the HTML table representing the playing field.
2. At every move in the game, the "*runner*" has to move one step in a direction at random, chosen between North, South, East, West. If the "*runner*" is already at the outmost location in the direction in which it ought to move, it just stays where it is; for example, if the "*runner*" is at a location on the leftmost column of the table, and the random direction where it ought to move is West, it remains where it is.
3. The *(x,y)* position of the "*runner*" character on the field can be any coordinate pair, with *x* in the range *[0 ... W-1]* where *W* is the number of *columns* in the table, and with *y* in the range *[0 ... H-1]* where *H* is the

number of *rows* in the table. Store the "*runner*" *(x,y)* position in two separate integer Javascript variables.
4. The *(x,y)* position of the "*runner*" character on the field also needs to be used to memorize the "*runner*" character as a 1-letter string in the Javascript 2D array representing the playing field, at its corresponding *column* and *row* in the array.

Task 2

The second task is to complete the input controls and game logic for moving the "*chaser*" character in the "*runner and chaser*" game, as presented at I211 lecture 22 pages 1 to 4 and page 6.

1. The "*chaser*" character in the game is represented by the "*number sign*" letter symbol "#". At any given time in the game, the "*chaser*" has to be shown in only one location in the HTML table representing the playing field.
2. At every move in the game, the "*chaser*" has to:
    a. *either* move one step in the direction chosen by the user by pressing one of the HTML buttons "N" *(for North)* "S" *(for South)* "E" *(for East)* "W" *(for West)*,
    b. *or* remain at its current location when the user presses the HTML button ".",
   and if the "*chaser*" is already at the outmost location in the direction in which it ought to move, it just stays where it is; for example, if the "*chaser*" is at a location on the rightmost column of the table, and the chosen direction where it ought to move is East, the "*chaser*" remains where it is.
   Also, update a Javascript variable keeping track of the total number of clicks.
3. The *(x,y)* position of the "*chaser*" character on the field can be any coordinate pair, with *x* in the range *[0 ... W-1]* where *W* is the number of *columns* in the table, and with *y* in the range *[0 ... H-1]* where *H* is the number of *rows* in the table. Store the "*chaser*" *(x,y)* position in two separate integer Javascript variables.
4. The *(x,y)* position of the "*chaser*" character on the field also needs to be used to memorize the "*chaser*" character as a 1-letter string in the Javascript 2D array representing the playing field, at its corresponding *column* and *row* in the array.

Task 3

The third task is to complete the game logic *model* and the *view*, as presented at I211 lecture 22 pages 7 to 10, and pages 1 to 4.

1. The game playing field *view* is to be implemented as an HTML table with H (height) rows, and W (width) columns in every row. Specify the viewing dimensions of each table row/column in HTML, either by hardcoding the size as *number of pixels* value in the `<tr>` and `<td>` HTML tags, or by assigning those values from Javascript.
2. The game playing field *model* is to be implemented in a 2D array, i.e. in Javascript as an array of arrays: the main array needs to contain as many items as the HTML table W (width) size, and each of these items needs to be an array containing as many items as the HTML H (height) size. To implement this array of arrays, you may use the MyArray2D() function as from lecture 22 notes. You have to obtain the HTML table W (width) size and the HTML H (height) size from the HTML table element.
3. Initialize the 2D array to contain one single space " " character in each array item at index *(x,y)*, with *x* in the range *[0 ... W-1]* where *W* is the number of *columns* in the HTML, and with *y* in the range *[0 ... H-1]* where *H* is the number of *rows* in the table. Then obtain the "*runner*" *(x,y)* position from the two separate integer Javascript variables as implemented in Task 1, and the "*chaser*" *(x,y)* position from the two separate integer Javascript variables as implemented in Task 2. Use these position pairs to write one "*at sign*" letter symbol "@" in the corresponding array item, and one "*number sign*" letter symbol "#" in the corresponding array item.
4. Write, into each cell of the HTML table, the character stored in the corresponding Javascript 2D array item (i.e. for each *i,j,* write into the j^{th} cell in the i^{th} row of the HTML table, as from I211 lecture 22 page 5).
5. At every move in the game, the 2D array needs to be updated thus:
    a. *erase* both runner and chaser characters from the 2D array, and write one single space " " character

in the two corresponding array items;
   b. *write* both runner and chaser characters into the 2D array: one "*at sign*" letter symbol "@" in the corresponding array item, and one "*number sign*" letter symbol "#" in the corresponding array item;
   c. refresh the content of the entire HTML table as from point (4) above;
   d. if the "*runner*" *(x,y)* position and the "*chaser*" *(x,y)* position coincide, display a Javascript alert "You won in *M* moves!", where *M* is the number of clicks on any of the input buttons, as from Task 2.

Task 4

Comment *every individual line* in your Javascript script code and your HTML tags, in the style of the example comments in the file `I211-lab-13-starting-text.html` for the `run()` function (the file `I211-lab-13-starting-text.html` is available in the *Oncourse Resources->Labs* folder).

1. including any already (as yet uncommented) provided code that you may use, e.g. if you use the `MyArray2D()` function from lecture notes, or if you use the `myArray()` variable, you need to explain how and why you use them, in comments above each line.
2. explain each Javascript reference to HTML elements, i.e.
   - whether any Javascript code outputs *to* HTML elements
   - whether any Javascript code receives input *from* HTML elements.
3. explain each HTML reference to Javascript code, i.e.
   - whether any HTML element *invokes* any Javascript function
   - whether any HTML element is *modified by* Javascript code.

***Bonus tasks*:**

i. (20 bonus points) add several "%" elements at random positions in the HTML table, with corresponding Javascript 2D array items, to implement *blocks* where neither one of the characters may move. These elements may remain at fixed (random) position during an entire game round.
ii. (20 bonus points) add one "!" element at random position in the HTML table: this element "defeats" either character that may be present at the same position, thereby concluding the game with the "*runner*" winning the game if the "*chaser*" is defeated by "!", or vice versa. This element needs to change its (random) position at every move in the game.

Assignment Submission:

When submitting the application, only one team member (of your choosing) is to submit all the code for the application, as well as a (common) team *README* file. All team members need to turn in their own (distinct) individual *README* file.

1. Include a team *README* plain-text file named `readme-a2-i211-`*`yourteamnumber`*`.text` in which you explain:
   - the parts of the assignment *your team* has completed
   - any extra functionality added to the code
   - any suggestions/enhancements you may have to improve the game
2. Include an individual *README* plain-text file named `readme-a2-i211-`*`yourusername`*`.text` in which you explain:
   - the parts of the assignment *you* have completed
   - the parts of the assignment *your team* has completed

    - any particularly clever code you added
    - (optional) any question you may have about the tasks or the provided template code
3. On the i211 / Spring 2014 site, turn in all your files by 11:59PM on Friday, April 25, 2014.
4. P.S.: *yourusername* should in fact be your username, not the word *yourusername*... and *yourteamnumber* should in fact be your team number, not the word *yourteamnumber*...

Include the following information as a `<!--comment-->` at the top of every HTML file you turn in:

```
<!-- your name (First, Last) -->
<!-- your IU email address -->
<!-- your I211 team number -->
<!-- the names of all your I211 team members -->
```

Each team member also needs to turn in the "I211 Student Team Feedback Form.doc" (which you can find in the *Oncourse Resources->Assignments* folder) about your experience in the student team, and turn it in on Oncourse with your Assignment 2.

Good luck!