Assignment 5

Due Date: Sunday, August 1st, 2023 11:59 pm.

Preparation:

Review the following topics

- Arrays
- Searching in an array
- Active Processing: setup() and draw()
- State-driven programming
- IF statement and nested IF
- Relational Boolean operators
- For loop
- User-defined functions with different return types

Notes:

- Name your sketches using your name, the assignment number, and the question number, exactly as in this example: DentStuwA5Q1 and DentStuwA5Q2
- The purpose of this program is to implement the Sudoku game and have a better understanding of arrays and searching for an element in an array.
- Assignments must follow the programming standards document.
- Make sure to use the processing console to show what the player types as the input.
- After the due date the dropbox will be closed and you will receive zero if you didn't submit on time.
- You can submit only one time to the dropbox and you cannot delete the older version.
- These assignments are your chance to learn the material for the exams. ***Code your assignments independently*.** We use software to compare all submitted assignments to each other, and pursue academic dishonestly vigorously.

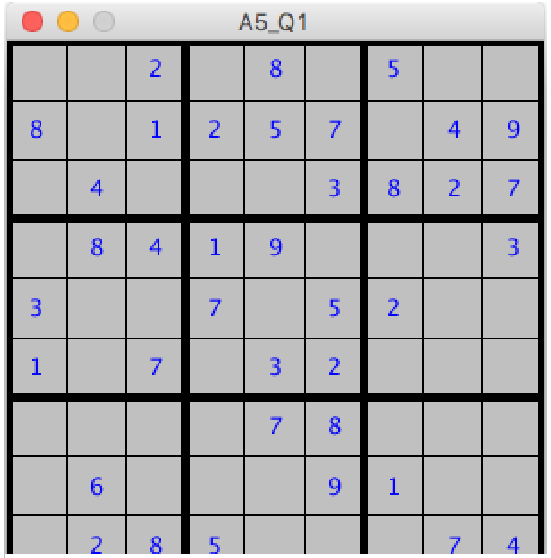
NOTE: For this assignment, you are allowed to use any of the approaches that you have learned in COMP 1010 and you should build up your solution based on your own understanding. If you use any other technique that is not covered in the COMP 1010 class, and you finish the assignment, no marks will be granted! For example, if you use multi-dimensional arrays instead of a 1 dimensional array, you will receive a **zero** as your mark for this assignment. If you are not familiar with the Sudoku and its rules, please refer to this link: https://en.wikipedia.org/wiki/Sudoku.

Q1: Draw a valid Sudoku look. (DentStewA5Q1.pde)

In this section, you will draw the look of a typical Sudoku. It will accept characters from the user and then draw those characters in the positions in the grid that are clicked on by the mouse. At the end of the assignment, the only valid inputs that should be accepted are digits between 1-9, but do not worry about that for this question! You will cover this in the next question. For this question, your program should be able to enter any character in the canvas, as long as there is an empty cell.

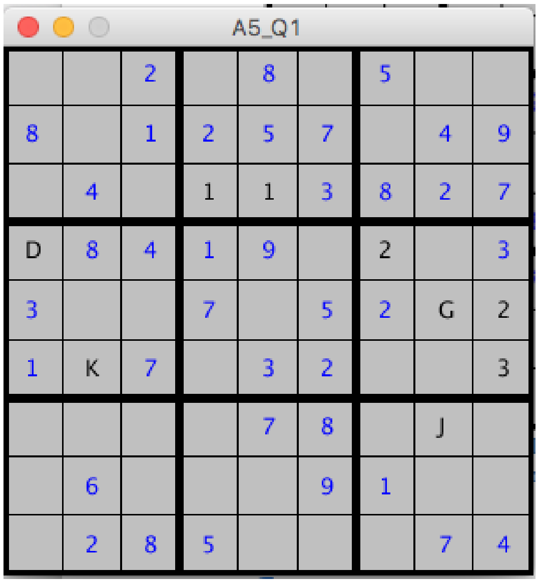
All the characters must be positioned at the center of a cell on the canvas. If the size of the canvas changes to other values between 100-900, with width and height always the same, the characters still have to be drawn at the center of the cells, and the text size should change to an appropriate size (not too big or too small).

In both Question 1 and Question 2, some numbers will already be in the grid when the program starts (place the blue numbers in the proper indexes of your array/s), and others will be entered by the user as in Question 1. Provide a suitable mechanism so that numbers that are there at the beginning will be drawn in a different color than the characters entered by the user. In the second question, you can delete a character that was enter by the user, but not the other numbers. Having a different colour makes it easy to spot the user-entered characters.
Make sure that your program draws a canvas very similar to **Figure 2**. This figure shows the Sudoku in a (300, 300) canvas.



		2		8		5		
8		1	2	5	7		4	9
	4				3	8	2	7
	8	4	1	9				3
3			7		5	2		
1		7		3	2			
				7	8			
	6				9	1		
	2	8	5				7	4

Figure 2: Sudoku in a (300, 300) canvas. The blue numbers are the given numbers to solve the puzzle.



		2		8		5		
8		1	2	5	7		4	9
	4		1	1	3	8	2	7
D	8	4	1	9		2		3
3			7		5	2	G	2
1	K	7		3	2			3
				7	8		J	
	6				9	1		
	2	8	5				7	4

Figure 1: Entered some random characters to the canvas. Remember that this is not valid for this game, we are just testing to see if the mouse clicks and array indexes are working.

Purpose of the code: When the code runs, type a character with the keyboard and then click on an empty cell with the mouse. If the cell is empty, draw the character. If the cell is not empty, you shouldn't be able to enter a character in the cell. By completing this question, your program should be able to map a mouse click on the canvas to the corresponding index in your array(s) (depends on your implementation).
Watch the video ***A5Q1_demo.MP4*** for an illustration. Submit your .pde file as the answer to this question into your dropbox.

Q2: A functional Sudoku with error detection (DentStewA5Q2.pde)

Save your DentStewA5Q1.pde program as DentStew_A5Q2.pde and then modify it. In this section, you will be completing your Sudoku which:

- Detects any incorrect character (not 1-9) and does not allow it to be entered.
- Detects any duplicates in a row, a column, or a 3x3 subgrid, notifies the player about the error, and does not accept the entry.
  - Use the JOptionPane class for showing the message.
- Can remove a user-entered character (but not one that was there at the beginning) and make the cell available again.
  - Use the key ‘c’ on the keyboard for doing this.
- Shows a winning message after the Sudoku is completely full, with valid entries.
  - Use the JOptionPane class for showing the message.

Watch the video ***A5Q2demo.MP4*** on moodle for an illustration.

Assignment 4 Grading Guide

NOTE: For all these questions setup() and draw() are required without which marks will not be awarded for that question.

Q1: A valid look of the Sudoku (20)

a) Drawing the grids with the bold lines to have 9 sub-grids **[2]**

b) Drawing all the provided numbers on the canvas at the center of each corresponding cell **[2]**

c) Locate a mouse click on the canvas and map the mouse click to the proper index of the array **[6]**
   - If the spot is filled by a character, it doesn’t change. (2)
   - If the spot is empty, enter the provided character to the canvas. (2)
   - Use a different color for the characters that are not provided at the beginning. (1)
   - Use the console to appropriately to communicate with the player. (1)

d) The program is able to insert a character **[1]**

e) Numbers are displayed at the center of cells. Change the canvas size from 100 to 900 to check this. **[3]**

f) Dividing a bigger task to smaller task through the whole program (usage of functions with different return types and different number of parameters/types). **[4]**

g) Programming standards **[2]**

Q2: A complete functional Sudoku (20)

a) Draws only digits between 1-9 in empty grids and not any other characters.

b) Accepts only unique numbers **[12]**
   - In a row (3)
   - In a column (3)
   - In a sub-grid (6)

c) Able to remove a user entered number and makes it available again for a new entry **[4]**
   - Removing (2)
   - Re-entering (2)

d) Display all the messages using JOptionPane as it is demonstrated on the video footage. **[2]**

e) Programming standards **[2]**

Total marks: 40

End of Assignment 5