

Design Overview for Sudoku Playing Game and Solution

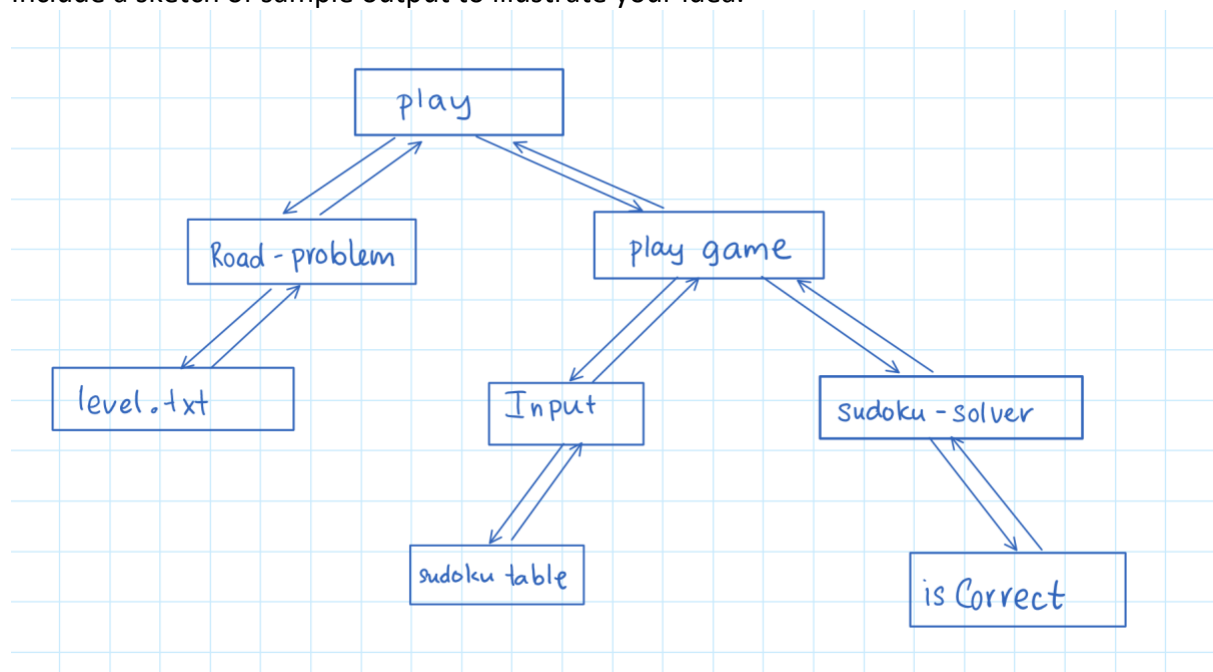
Name: Viet Dung Tran
Student ID: 103486496

Summary of Program

Describe what you want the program to do... one or two paragraphs.

The custom project is about sudoku gameplay which allow user to play the game. The users are required to click the number below the sudoku table to add the number inside the sudoku table. When the users cannot solve the problem, they can see the solution by clicking the solution button. The problem will be loaded from a txt file, and there will be different problem at different level of difficulty to solve. The computer will calculate the time user spending on solving the problem and come up with scores

Include a sketch of sample output to illustrate your idea.



Required Data Types

Describe each of the records and enumerations you will create using the following table (one per record).

Table 1: <<record name>> details

Field Name	Type	Notes
------------	------	-------

Node	Node	The node of CircularLinkedList
CircularLinkedList	Circular Linked List	CircularLinkedList
Button	Button location	Record of buttons
Square	Square of table	The square of the table
SudokuTable	Sudoku table	The Sudoku table records
Buttons for each scene	Hash table	Series of buttons in each scene
Problems	Circular Linked List	Series of problems read from text file
set	Set	Set to save unique data

Table 2: <<enumeration name>> details

Value	Notes
Zorder	Display the element in background Player or the top interface
grid of sudoku table	The current sudoku table Two dimensional arrays of problem's grid

...

Overview of Program Structure

List the main functions/procedures you are going to need to create this program. For each function/procedure provide its name and a brief description of what it will do.

Don't spend too long on this at this stage. Focus on the main things you think you are likely to need and you can build on this as your program develops.

The main function and procedures:

The program has four main functions: initialize, update, draw, button_down basically in a Gosu program

- Initialize:
 - initialize_start: initialize variables when user in start scene
 - initialize_gameplay: initialize variables when user in gameplay
 - initialize_tutorial: initialize variables in tutorial scene
 - initialize_difficulty: initialize variables in difficult scene
- Update:
 - update_gameplay: every function run every time during gameplay scene
- Draw:
 - draw_gameplay: draw every objects in gameplay scene
 - draw_start: draw every objects in start scene
 - draw_difficulty: draw every objects in difficulty scene
 - draw_tutorial: draw every objects in tutorial scene
- Button_down:

button_down_gameplay: handle button down action in gameplay scene
 button_down_start: handle button down action in start scene
 button_down_tutorial: tutorial: handle button down action in tutorial scene
 button_down_difficulty: handle button down action in difficulty scene

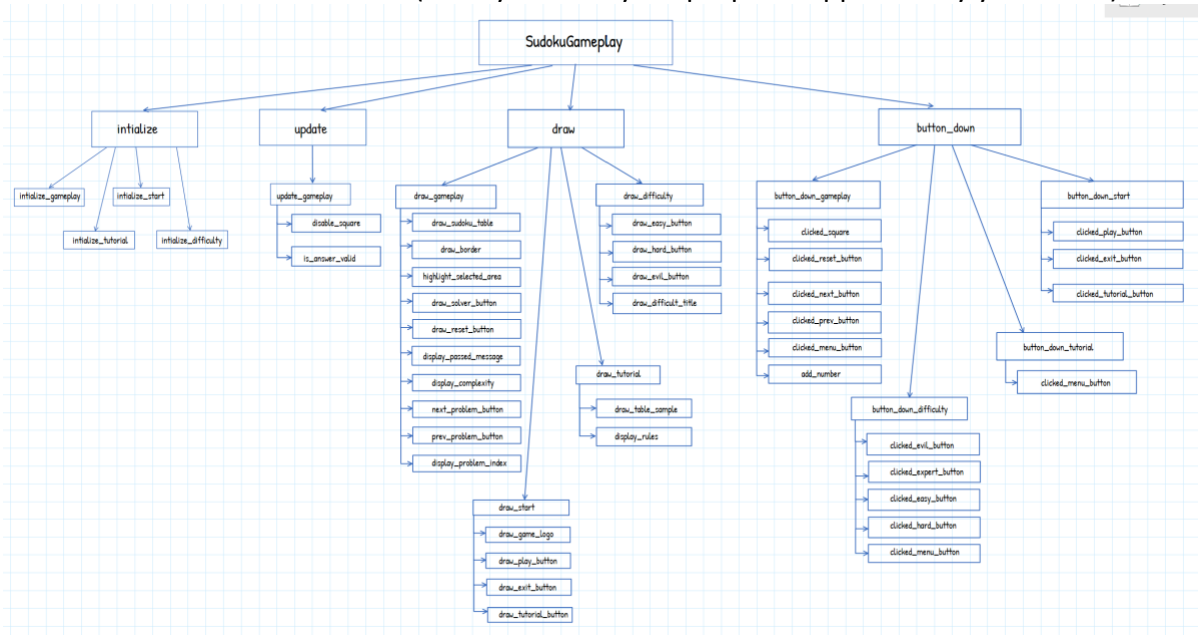
- Sudoku solver:

sudoku_solution: return true if the solution has been found by backtracking

is_empty_square: find empty square from the problem if there is no empty square
 return true

is_safe: checking if the assigned value have any duplication within its row, column or box of 3x3

Include a structure chart (once you have your proposal approved by your tutor)





Complexity: Evil

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

Reset

Solver

Menu

<< Prev

1 / 7

Next >>