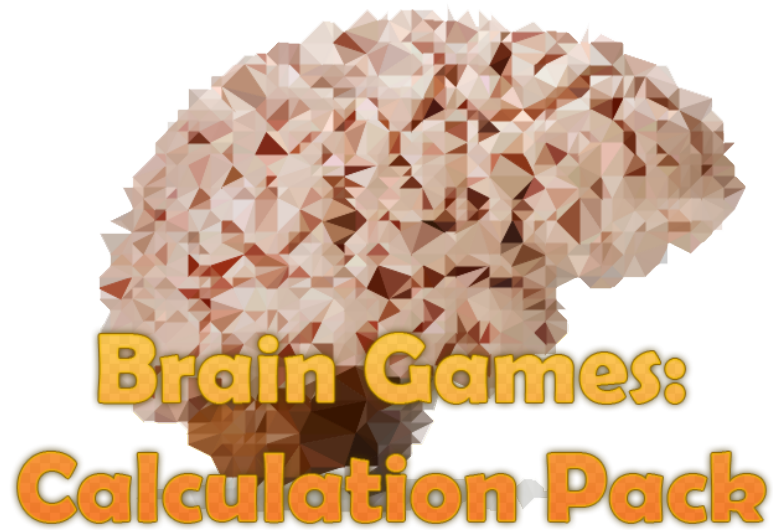


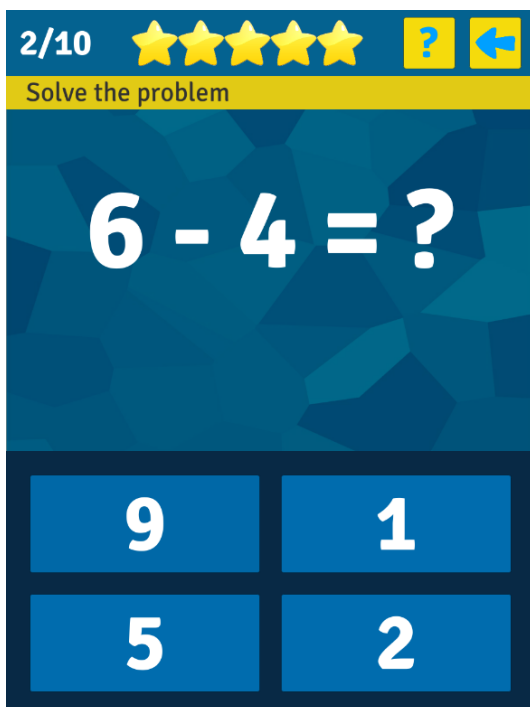
## Brain Games: Calculation Pack



## I. Project description

The project is a set of mathematical games in which you must choose the right answers. The program includes games:

1. **Calculation** – the objective is to find the correct answer. Choose the correct answer from four options. For example, the correct answer for  $6+2$  is 8.



2. **Choose a symbol** – the goal is to choose a symbol which will complete the problem and solve it. For example to solve the problem  $2 ? 3 = 5$  you need to choose the '+' symbol.

1/10 ★★★★★ ? ←

Choose a symbol to complete the problem.

$5 ? 5 = 10$

+	+
+	-

The screenshot shows a game interface with a blue background. At the top, it displays '1/10', five yellow stars, a question mark icon, and a back arrow icon. Below this is a yellow banner with the text 'Choose a symbol to complete the problem.' The main area shows the equation '5 ? 5 = 10' in large white text. At the bottom, there is a 2x2 grid of blue buttons with white symbols: the top row contains two '+' signs, and the bottom row contains a '+' sign and a '-' sign.

3. **Choose a number** – fill in a missing number to solve the problem. In the problem  $3 + ? = 5$  the missing number is 2.

1/10 ★★★★★ ? ←

Choose a number to complete the problem.

$? + 9 = 12$

3	4
19	7

The screenshot shows a game interface with a blue background. At the top, it displays '1/10', five yellow stars, a question mark icon, and a back arrow icon. Below this is a yellow banner with the text 'Choose a number to complete the problem.' The main area shows the equation '? + 9 = 12' in large white text. At the bottom, there is a 2x2 grid of blue buttons with white numbers: the top row contains '3' and '4', and the bottom row contains '19' and '7'.

4. **Counting pictures** – you have to solve a problem where numbers are represented by different pictures. The number of pictures represents the numeric value. For example 3 apples + 4 apples equals 7.

1/10 ★★★★★ ? ←

Solve the problem.

○ + ○○○

7	6
3	9

5. **Moving numbers** – the goal is to solve the problem by calculating the correct answer. If there is a subtraction or division sign, the higher number is always used first. So if you see the numbers 5 and 30 and the division sign, the correct answer is 6

1/10 ★★★★★ ? ←

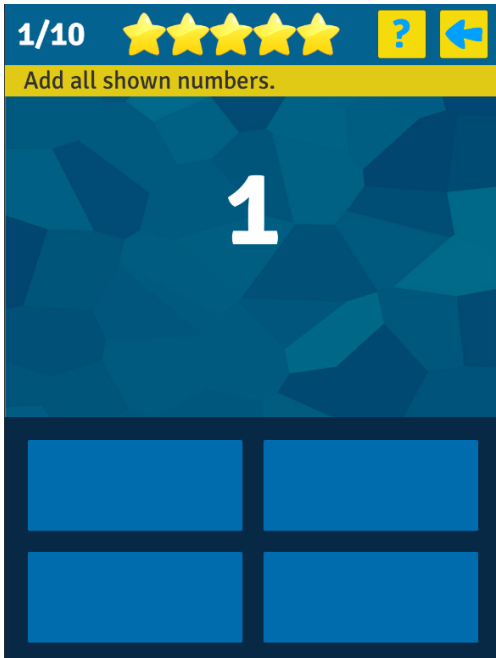
Solve the problem.

3 4

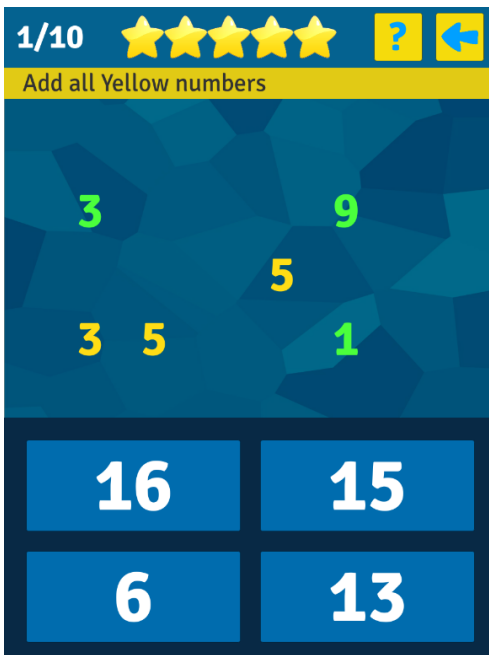
+

12	5
9	7

6. **Flashing numbers** – first you will see a set of flashing numbers which you have to add together. Then choose the correct answer from four offered options.



7. **Colorful numbers** – you have to add together all numbers of the same color. The color is specified in the assignment. Be careful, the color changes.



8. **Shining numbers** – include only those numbers in the grid that are highlighted.

1/10 ★★★★★ ? ↩

Add all numbers in highlighted boxes.

9	8	7
6	5	4
3	2	1

18      11

12      2

9. **Min and Max** – first determine the highest and the lowest number of the four shown and then solve the problem. If there is a subtraction or division sign, always use the higher number first.

1/10 ★★★★★ ? ↩

Use the highest and lowest numbers.

3

8 + 4

2

2      1

9      10

10. **Group counting** – add all the shown numbers together and then choose the correct number out of the four options.

1/10 ★★★★★ ? ↩

Add numbers in a group.

2 2

2 2

12	8
6	10

11. **Three numbers** – solve a problem with three numbers.

1/10 ★★★★★ ? ↩

Solve the problem.

$9 + 18 + 13$   
 $= ?$

43	40
45	49

12. **Equation** – fill in the first and the second number to get the correct equation.  
For example if the problem is  $? - ? = 8$  and the numbers to choose from are 1 5 9 0,  
choose 9 first and then 1.

1/10 ★★★★★ ? ↩

Fill in the numbers to complete the equation.

$$[ ] + [ ] = 22$$

17	5
17	24

13. **Choose 2 symbols** – choose the correct symbols to complete the equation. For example  $4 ? 4 ? 1 = 1$  the correct symbols are '-' and '+'

1/10 ★★★★★ ? ↩

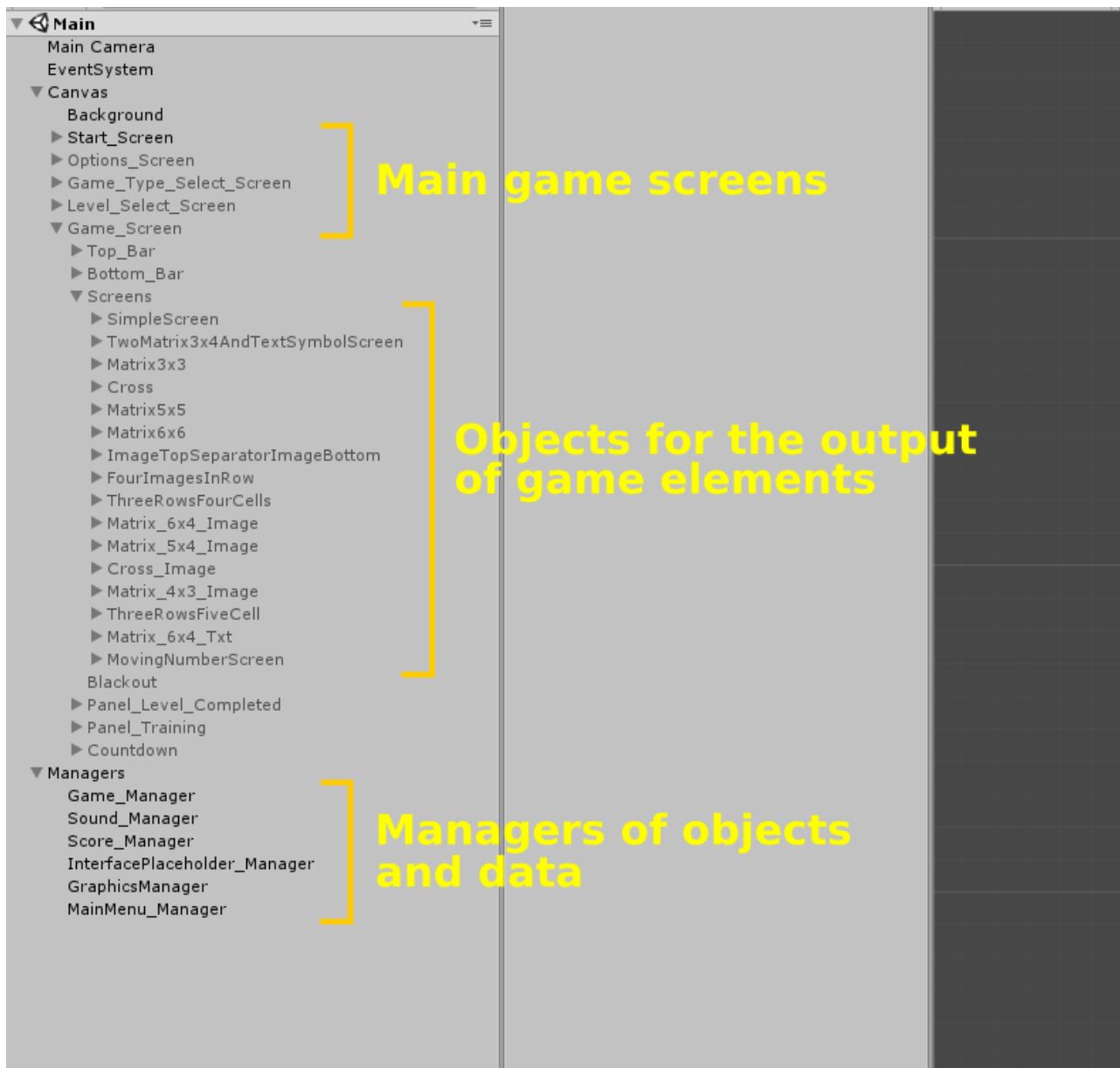
Choose 2 symbols to get a correct answer.

$$14 [ ] 6 [ ] 4 = 4$$

++	+-
-+	--

## II. Description of the elements on the stage

The scene includes all the necessary elements for the work of the game - game screens for each section of the program, interface elements, objects for displaying game elements, as well as resource managers.

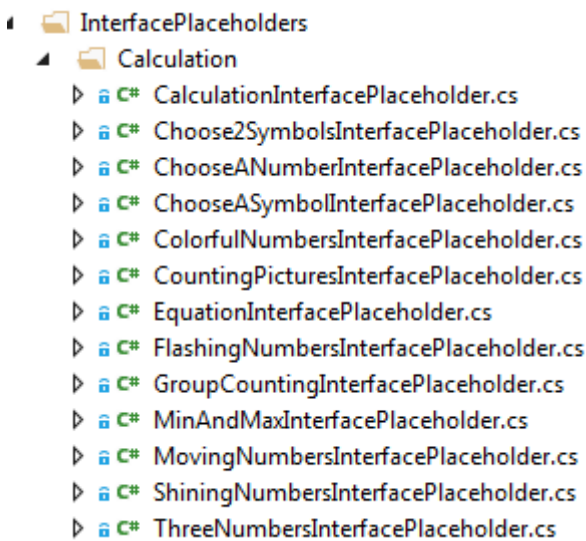




### III. Description of scripts, structure of work

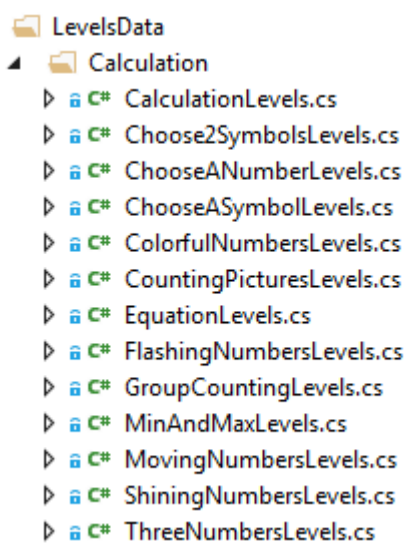
The logic of the program is divided into several main parts:

**InterfacePlaceHolders** – it includes scripts for managing the main data of each game, obtaining information about the current level and placing the objects on the screen.



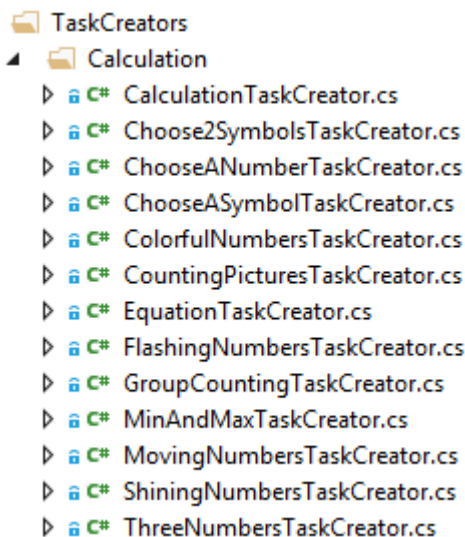
*Pic. InterfacePlaceHolders*

**LevelsData** – the scripts in this section contain information about the main parameters of each level for each game. Include both mandatory parameters (totalLevelsInGame, totalStagesInGame), and specific to a particular game.



*Pic. LevelsData*

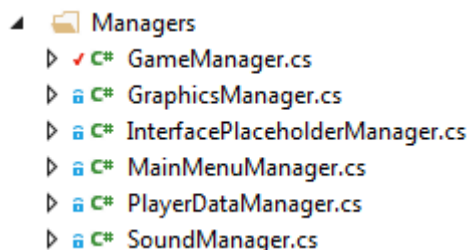
**TaskCreators** – are responsible for preparing the basic data for the game, based on the LevelsData data, that is, they select specific values for the game parameters for each level: the values of the operands in the expression, the choice of the operator, the calculation of the correct answer, and the creation of incorrect answers.



*Pic. TaskCreators*

**Managers** – Scripts for managing objects, information and the state of the program. The GameManager script is basic, includes links to most game objects, and also implements the transition between game screens and starts the process of creating the selected game. Tasks of other managers :

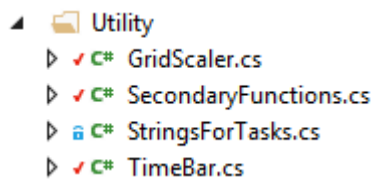
- To provide access to graphic resources of the game (GraphicsManager)
- Access to game creation scripts (InterfacePlaceholderManager)
- Control of the main menu (MainMenuManager)
- Record and read player results - points, open levels (PlayerDataManager)
- Management of music and sounds (SoundManager).



*Pic. Managers*

**Utility** – additional auxiliary functions used in different parts of the program :

- GridScaler – scaling objects included in an object with Grid Layout Group component;
- SecondaryFunctions – realizes the functionality of creating random integers in a given range, creating incorrect answers, creating statements for expressions, obtaining colors for game objects, mixing arrays (integer), converting game data ;
- StringForTasks – includes text data that fill the game interface - the name of the selected game, a short description, help text, messages about the end of the game .
- TimeBar - the code contains functions for controlling the slider, which displays the available time remaining for the game without losing.



*Pic. Utility*