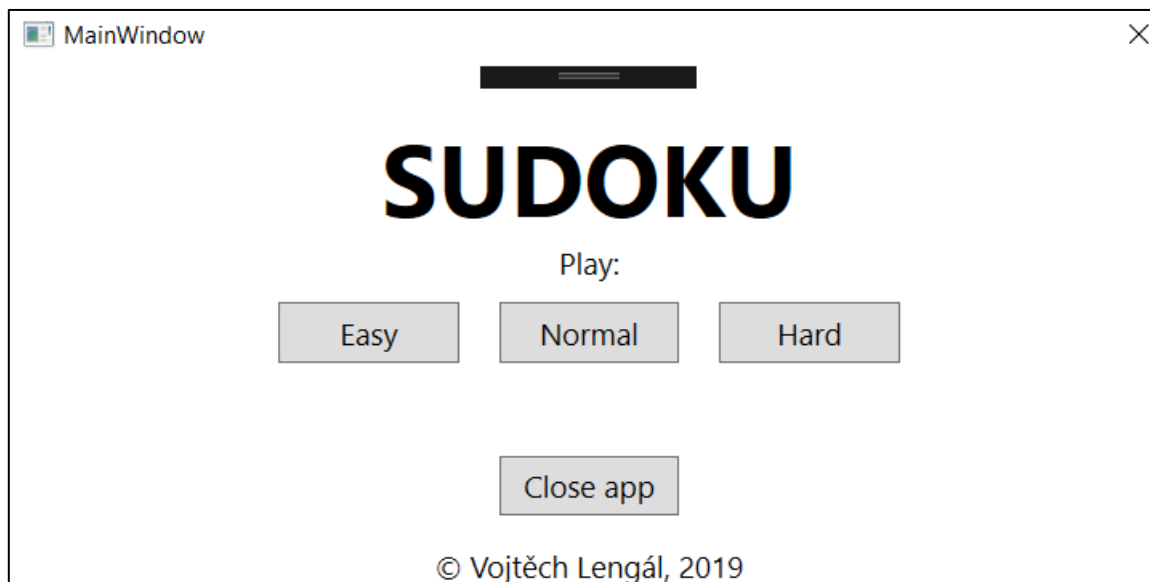


SUDOKU - dokumentace

Vojtěch Lengál

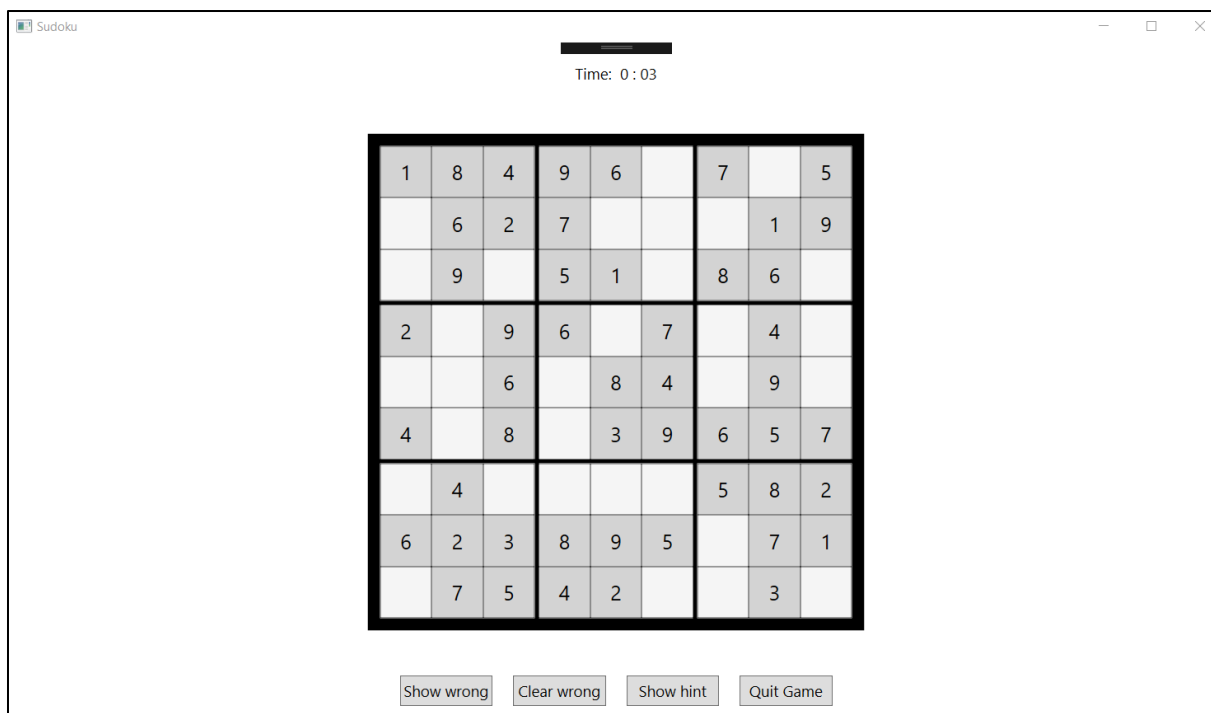
Ovládání

Po spuštění aplikace se zobrazí výchozí obrazovka (viz. Obrázek 1).



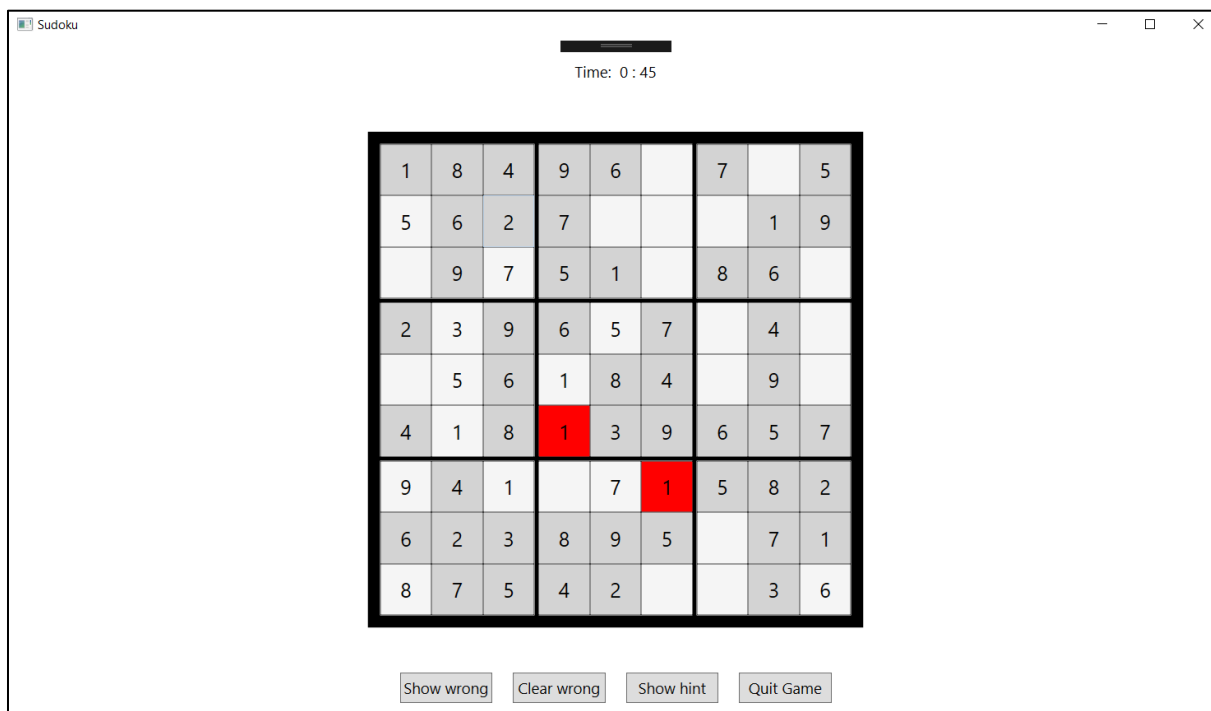
Obrázek 1

Uprostřed obrazovky je menu s tlačítky. Kliknutím na tlačítko „Easy“, „Normal“ nebo „Hard“ spustíte příslušný herní mód. Pokud chcete aplikaci ukončit, klikněte na tlačítko „Close App“. Po spuštění se zobrazí vygenerované sudoku (viz. Obrázek 2).



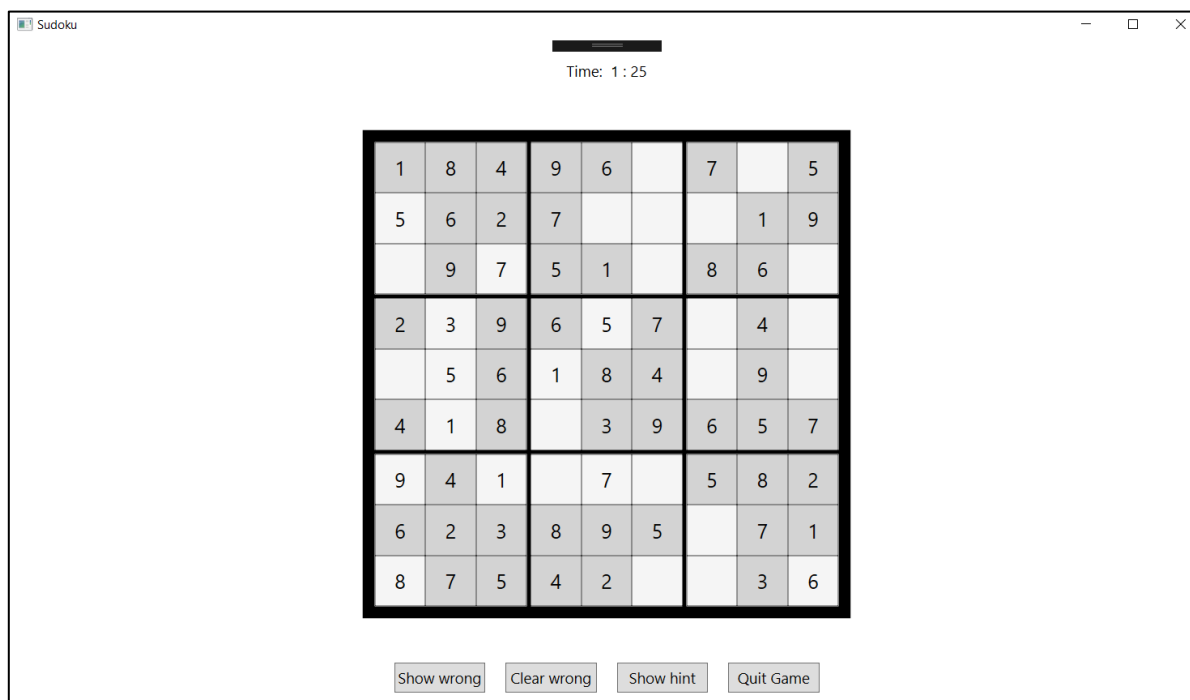
Obrázek 2

Políčka s šedým pozadím jsou nastavená defaultně a jejich hodnotu nelze měnit. Do bílých polí hráč doplňuje čísla. Hodnotu v poli změníte tak, že na příslušné pole kliknete, zapíšete novou hodnotu a změnu potvrdíte stisknutím klávesy Enter nebo kliknutím do jiného pole. V dolní části okna jsou 4 tlačítka. Tlačítko „Show Wrong“ zvýrazní červeně pole, která mají špatnou hodnotu (rozdílnou oproti vzorovému řešení) – viz. Obrázek 3



Obrázek 3

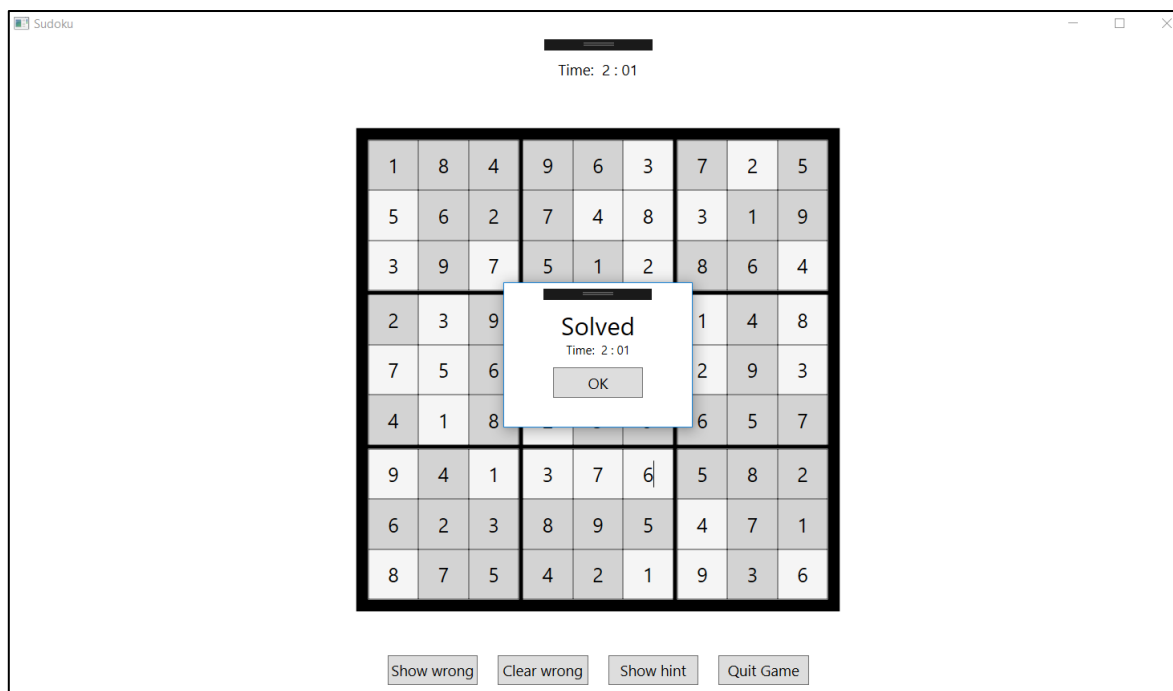
Tlačítko „Clear Wrong“ vymaže hodnotu špatně vyplněných polí (viz. Obrázek 5 – situace po stisknutí tlačítka na herní desce z Obrázku 4).



Obrázek 4

Stisknutím tlačítka „Show hint“ využijete nápovědu – počítač doplní 1 hodnotu podle vzorového řešení. Pro ukončení hry a návrat do hlavního menu stiskněte tlačítko „Quit game“.

Pokud sudoku správně vyřešíte, tak se zastaví časomíra a ukáže se pop-up okno (viz. Obrázek 5).



Obrázek 5

Stisknutím tlačítka „OK“ a následně „Quit game“ se dostanete zpět to hlavního menu.

Popis aplikace

Jedná se o simulaci hry Sudoku.

Aplikace je naprogramovaná v jazyce C# s využitím frameworku WPF. Při vývoji jsem použil MVVM architekturu, třídy jsou rozdělené do složek *Models*, *ViewModels*, *Views*. Modely jsou třídy, které vykonávají nějaké výpočty nebo pracují s daty. Jako View se označuje prezentační část aplikace. Ve WPF jde o XAML soubory (tzn. v našem případě jednotlivá okna) a jejich Code Behind, což je třída, která přísluší danému oknu. ViewModel slouží k předávání dat mezi Models a Views. Ke každému View přísluší standartně 1 ViewModel. Ten obsahuje data, která jsou potřeba zobrazit v daném okně.

Ve složce *Models* je navíc speciální třída *ObservableObject*, která implementuje rozhraní *INotifyPropertyChanged*, a umožňuje tak v programu použít two-way data binding. Frontend je napsaný v jazyce XAML, což je modifikace jazyka XML (XAML se ve frameworku WPF používá pro layout defaultně).

Hra má 3 obtížnosti – lehkou, normální a těžkou.

Hrací plán 9x9 je reprezentovaný třídou *Board*, ta obsahuje 9 čtverců 3x3 (třída *SudokuSquare*). Každý čtverec velikosti 3x3 obsahuje 9 políček (třída *SudokuCell*). V projektu je dále třída *SudokuGenerator*, která se používá k vygenerování sudoku podle zadané obtížnosti. Postup je následující: máme validně vyplněné 9x9 sudoku, to následujícími úpravami editujeme:

- Prohození 2 řádků nebo sloupců sudoku (o 9 číslech), které se nacházejí ve stejných 3x3 čtvercích (viz. obrázek)
- Prohození 2 řádků / sloupců s 3x3 čtverci (viz. obrázek)
- Transpozice celé desky (prohození řádky - sloupce)

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

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

Těmito úpravami se správnost řešení nezmění. Poté postupně odebíráme hodnoty, hodnotu odebereme vždy, pokud je jednoznačně určena (tzn. na ostatních prázdných pozicích v řádku, sloupci, popř. ve čtverci daná hodnota nemůže být, nebo je daná hodnota jediná možná, která může být na dané pozici) Tímto způsobem vygenerujeme validní sudoku.

Implementace tohoto postupu se nachází v metodě *GetValidCompleteSudoku*, která vygeneruje validní sudoku. Poté se v kódu volá metoda *RemoveValues*, ta vymaže daný počet políček. Počet políček, jenž je potřeba vymazat, se určí podle zvolené obtížnosti Sudoku.

V aplikaci se dále nachází třída *SudokuManager*, která slouží pro ostatní operace se sudoku jako např. hledání špatně vyplněných polí (metoda *ShowWrongValues*), mazání jejich hodnoty (metoda *ClearWrongValues*), atd. Pomocí metody *IsSolved* v programu kontroluji, jestli je sudoku již vyřešené. Kontrola probíhá tak, že vyberu všechny řádky, sloupce a 3x3 čtverce hrací plochy a následně zkontroluji, jestli je v každém seznamu o 9 políčkách každá hodnota právě jednou.