

Programátorská dokumentace

Název programu: Sudoku

Úvod

Tento program umožňuje hráčům hrát klasickou hru Sudoku, která spočívá v vyplnění 9x9 mřížky čísly tak, aby byla splněna pravidla – každé číslo od 1 do 9 musí být právě jednou ve všech řádcích, sloupcích a v každém 3x3 bloku.

Dokumentace popisuje funkcionalitu a implementaci programu pro hraní a řešení hry Sudoku. Program je napsán v jazyce C# s využitím platformy Windows Forms.

Struktura kódu

Program obsahuje třídu Form1, která představuje hlavní formulář aplikace. Tato třída obsahuje metody pro práci s herním polem, generování her a interakci s uživatelem.

Metody a funkce

1. LinkTextBoxesToArray

Metoda pro propojení textboxů ve vizuálním ovládacím prvku TableLayoutPanel s dvourozměrným polem textBoxes.

Jejím účelem je přiřadit každý TextBox z vizuálního prvku TableLayoutPanel do příslušné pozice v poli textBoxes.

Implementace: Metoda prochází všechny řádky a sloupce vizuálního kontejneru TableLayoutPanel. Pro každý řádek a sloupec získá ovládací prvek. Tento prvek je typu TextBox a je přiřazen do příslušného místa v poli textBoxes.

2. SolveSudoku

Tato metoda slouží k řešení hry Sudoku pomocí rekurzivního algoritmu zpětného sledování. Metoda se pokouší doplnit čísla do prázdných buněk herního pole tak, aby splňovala pravidla hry Sudoku.

Metoda má výstupní typ bool, který indikuje úspěšnost řešení. Pokud se podaří najít řešení, metoda vrátí true, jinak vrátí false.

Tato metoda se volá po generování nové hry nebo při řešení již existující hry.

Implementace: Metoda začíná tím, že hledá první prázdnou buňku v herním poli pomocí metody FindEmptyCell. Pokud žádná prázdná buňka není nalezena, tedy herní

pole je plné, metoda vrátí true, což znamená úspěšné řešení hry. Pokud je nalezena prázdná buňka, metoda prohází čísla od 1 do 9 a postupně zkouší doplnit čísla do buňky. Pro každé číslo z permutace provádí následující kroky:

1.Ověří platnost tahu pomocí metody IsValidMove, která kontroluje, zda lze číslo umístit na danou pozici v herním poli.

2.Pokud je tah platný, doplní číslo do buňky, aktualizuje herní pole a pokračuje rekursivním voláním metody SolveSudoku pro další buňku.

3.Pokud se rekursivní volání vrátí s výsledkem true, znamená to, že hra je úspěšně vyřešena, a metoda SolveSudoku také vrátí true.

4.Pokud se rekursivní volání vrátí s výsledkem false, znamená to, že dané číslo není správné a metoda se vrátí do původního stavu a zkusí další číslo.

3. FindEmptyCell

Tato metoda slouží k nalezení první prázdné buňky v herním poli. Prázdná buňka je taková buňka, která neobsahuje žádné číslo a je určena pro vyplnění hráčem nebo algoritmem řešení.

Metoda má dva výstupní parametry: row a col, které reprezentují řádek a sloupec první nalezené prázdné buňky. Pokud nebyla nalezena žádná prázdná buňka, hodnoty row a col budou nastaveny na -1.

Implementace: Metoda postupně prochází všechny buňky herního pole. Pokud narazí na prázdnou buňku vrátí true a výstupní parametry, pokud nebyla nalezena žádná, vrátí false.

4. IsValidMove

Tato metoda ověřuje, zda je daný tah (umístění čísla) na zadané pozici [řádek, sloupec] v herním poli platný. Platný tah znamená, že číslo lze umístit do buňky na dané pozici tak, aby nedošlo ke konfliktu s čísly v stejném řádku, sloupci nebo 3x3 bloku

Parametry: row (integer): Řádek, ve kterém se nachází testovaná buňka; col (integer): Sloupec, ve kterém se nachází testovaná buňka; num (integer): Číslo, které se má umístit do buňky.

Metoda vrátí true pokud je tah platný a číslo lze umístit do buňky na zadané pozici nebo false pokud je tah neplatný a číslo nelze umístit do buňky na zadané pozici.

Implementace: Metoda provádí několik kontrol, aby ověřila, zda lze číslo umístit na pozici [row, col] v herním poli:

1. Kontrola řádku a sloupce: Prochází všechny buňky v daném řádku a sloupci a zjišťuje, zda se číslo již nevyskytuje. Pokud ano, tah není platný.

2. Kontrola 3x3 bloku: Vypočítá počáteční pozici 3x3 bloku obsahující buňku na pozici [row, col]. Prochází všechny buňky v tomto bloku a zjišťuje, zda se číslo již nevyskytuje. Pokud ano, tah není platný.

5. start_but Click

Tato metoda je vyvolána po stisknutí tlačítka "Start" na formuláři a slouží k zahájení nové hry Sudoku. Při stisku tohoto tlačítka je možné zvolit obtížnost hry a vygenerovat novou hru s odpovídajícím množstvím odstraněných prvků na herním poli.

Implementace: Metoda zjišťuje, která z radiových tlačítek (Easy, Medium, Hard) je zaškrtnuta, a na základě toho určuje počet čísel k odstranění pro danou obtížnost. Pokud není zaškrtnutá, je nastavena na střední. Metoda volá metodu ClearSudokuBoard k vynulování všech políček a metodu GenerateSudoku s počtem čísel k odstranění získaným z volby obtížnosti. Tím dojde k vygenerování nové hry na herní desce. Po vygenerování hry zavolá metodu DisplaySudokuBoard pro aktualizaci vizuálního zobrazení herního pole.

6. Solution_Click

Tato metoda je vyvolána po stisknutí tlačítka "Solution" na formuláři a slouží k nalezení a zobrazení řešení aktuální hry Sudoku.

Implementace: Po stisknutí tlačítka "Solution" je volána metoda SolveSudoku, která se pokusí vyřešit aktuální hru.

7. ClearSudokuBoard:

Tato metoda slouží k vymazání obsahu herního pole a připravení ho pro novou hru.

Implementace: Metoda iteruje přes všechny buňky herního pole a nastavuje jejich hodnotu na 0. Stejným způsobem jsou všechny textboxy na herním poli vyčištěny.

8. DisplaySudokuBoard

Tato metoda slouží k vizuálnímu zobrazení obsahu herního pole v textbozech na herní desce. Po zavolání této metody jsou hodnoty v herním poli zobrazeny v příslušných textbozech na herní desce.

Implementace: Metoda iteruje přes všechny buňky herního pole. Pro každou buňku herního pole je zkontrolováno, zda obsahuje nenulovou hodnotu. Pokud ano, je tato hodnota zobrazena v příslušném textboxu a nastavují se atributy readonly a enabled tak, aby uživatel nemohl hodnotu měnit. Pokud je hodnota buňky nulová, je text v textboxu smazán a atributy readonly a enabled jsou nastaveny pro možnost zadání hodnoty.

9. GenerateSudoku

Tato metoda slouží k generování nové hry Sudoku s určitým počtem odstraněných prvků podle zvolené obtížnosti. Metoda zajišťuje, že vygenerovaná hra má jednoznačné řešení.

Parametry: countRemove: Počet čísel k odstranění z kompletního řešení hry podle obtížnosti.

Implementace:

1. Nejprve je zavolána metoda SolveSudoku() pro vyřešení kompletní hry Sudoku. Tím získáme kompletní řešení, které bude sloužit jako základ pro generování nové hry.

2. Následně se vytvoří prázdný seznam cellsToRemove, který bude obsahovat buňky, z nichž budou odstraněna čísla. Do této části probíhá proces odstraňování čísel z herního pole. V cyklu se náhodně vybírají buňky podle jejich řádku a sloupce. Pokud buňka dosud není v seznamu cellsToRemove, dočasně se uloží její hodnota, číslo v buňce se nastaví na nulu a následně se ověří, zda odstranění čísla z této buňky vytváří stále jednoznačné řešení. Pokud ne, číslo se vrátí zpět do buňky. Pokud odstranění čísla vytváří jednoznačné řešení, buňka se přidá do seznamu cellsToRemove.

3. Nakonec se pro každou buňku v seznamu cellsToRemove odstraní číslo z herního pole.

10. HasUniqueSolution:

Tato metoda slouží k ověření, zda aktuální herní pole má jednoznačné řešení.

Výstup je buď true: pokud aktuální herní pole má jednoznačné řešení nebo false: pokud aktuální herní pole nemá jednoznačné řešení.

Implementace: Metoda využívá metodu CountSolutions() pro zjištění počtu všech možných řešení aktuálního herního pole. Pokud počet řešení je roven 1, pak aktuální herní pole má jednoznačné řešení.

11. CountSolutions:

Tato metoda slouží k počítání všech možných řešení hry pro aktuální herní pole.

Výstupem je celkový počet všech možných řešení hry pro aktuální herní pole.

Implementace: Metoda prochází jednotlivé buňky herního pole a pokouší se do nich umístit čísla od 1 do 9. Při každém umístění čísla rekurzivně volá samu sebe pro další buňky a pokračuje ve snaze dosáhnout dalších řešení. Pokud během tohoto procesu narazí na situaci, kdy nelze umístit žádné další číslo, vrátí se zpět a zkouší jinou možnost.

12. checkButton_Click:

Tato metoda slouží k ověření správnosti vyplněných čísel ve hře a zobrazení výsledku hráči.

Implementace: Metoda projde všechny vyplněné buňky herního pole a porovnává jejich hodnoty s kontrolním polem checkBoard. Pokud jsou hodnoty shodné, znamená to, že hráč správně vyplnil danou buňku, a tato buňka je uzamčena pro další úpravy. Pokud jsou hodnoty různé, zvýrazní se text buňky červenou barvou, aby bylo jasné, že hráč udělal chybu. Zároveň metoda kontroluje, zda jsou všechny buňky vyplněny. Pokud jsou všechny buňky vyplněny a správně, zobrazí se hráči gratulace.

13. TextBox_KeyPress

Tato metoda slouží k ovládání chování textových polí při stisku kláves na klávesnici.

Implementace: Metoda zajišťuje, že do textových polí je možné vkládat pouze číslice. Pokud je stisknutá jiná klávesa, metoda nastaví vlastnost Handled události na true, čímž zamezí vkládání dané klávesy do textového pole.

14. TextBox_TextChanged

Tato metoda slouží k ověření a omezení hodnot vložených do textových polí.

Implementace: Metoda provádí kontrolu obsahu textového pole. Pokud je v textovém poli obsažena neprázdná hodnota, metoda převede na celé číslo. Pokud je hodnota v rozsahu od 1 do 9, ponechá se tato hodnota v textovém poli. V opačném případě se hodnota v textovém poli smaže.

15. InitializeTextBoxes

Tato metoda slouží k propojení událostí KeyPress a TextChanged s textovými poli ve vizuálním ovládacím prvku TableLayoutPanel.

Implementace: Metoda projde všechna textová pole v rámci vizuálního ovládacího prvku `TableLayoutPanel` a propojí je s obslužnými rutinami událostí `KeyPress` a `TextChanged`. Každé textové pole je propojeno s těmito dvěma událostmi, což umožňuje provádět omezení vstupu na čísla a zároveň kontrolu platnosti vložených hodnot

16. `AttachEnterEventHandlers`

Tato metoda slouží k propojení události `Enter` s textovými poli ve vizuálním ovládacím prvku `TableLayoutPanel`.

Implementace: Metoda projde všechny ovládací prvky uvnitř vizuálního ovládacího prvku `TableLayoutPanel` a pokud se jedná o textové pole, připojí k němu událost `Enter`. Tato událost může například změnit vizuální vzhled textového pole, pokud do něj hráč vstoupí.

17. `TextBox_Enter`

Metoda `TextBox_Enter` je událostí `Enter` pro textová pole.

Implementace: Metoda `TextBox_Enter` slouží k resetování vizuálního vzhledu textového pole v okamžiku, kdy hráč do něj vstoupí. Konkrétně se zde provádí změna barvy textu textového pole zpět na standardní černou barvu, pokud byla předtím změněna na červenou jako nalezení chyby.

Závěr

Tato programátorská dokumentace poskytuje podrobný přehled o implementaci a funkcionalitách aplikace `SudokuGame`, což je program určený pro hraní a řešení hry `Sudoku`. Dokumentace obsahuje detailní popis tříd, metod a funkcí programu, které umožňují interakci s uživatelem, generování her různých obtížností, řešení her a kontrolu správnosti řešení.

Program `SudokuGame` je napsán v jazyce `C#` a využívá `Windows Forms` pro tvorbu grafického uživatelského rozhraní. Hlavním cílem aplikace je umožnit uživatelům hrát a řešit hru `Sudoku`, a to včetně možnosti generování nových her s různými obtížnostmi a ověření správnosti zadaného řešení.