

ISI – 2. Zadanie

Sudoku - Vladislav Močilenko, Matúš Kuriľak

Sudoku je hra ktorá sa odohráva na hracom poli, často o rozmeroch 9x9 (najmenších štvorcov herného poľa) v ktorej je cieľom vyplniť každý riadok, stĺpec a každý hrubou čiarou vyznačený štvorec o rozmeroch 3x3 (najmenších štvorcov herného poľa), číslami od 1 až po 9 v ľubovoľnom poradí tak, aby sa v žiadnom zo stĺpcov, riadkov, ani 3x3 štvorcov herného poľa žiadne z čísel neopakovalo. V počiatočnej situácii na začiatku hry je časť čísel vždy predvyplnená, v súlade s týmito pravidlami.

1. Backtracking (Spätné sledovanie)

Ako funguje: Algoritmus prechádza hraciu plochu bunka po bunke, skúša čísla od 1 po N (veľkosť sudoku). Ak číslo vyhovuje pravidlám sudoku, vloží ho a pokračuje ďalej. Ak narazí na problém, vráti sa späť (backtrack) a skúša ďalšie možnosti.

Výhody: Jednoduchý a spoľahlivý prístup, ktorý vždy nájde riešenie, ak existuje.

Nevýhody: Môže byť pomalý, pretože skúša všetky možnosti (bruteforce).

Minimálny počet krokov: 227 (Stav 4)

Maximálny počet krokov: 4483 (Stav 7)

Priemerný počet krokov: 1203.1

2. DFS (Hĺbkové prehľadávanie - Depth First Search)

Ako funguje: Používa zásobník (stack) na prehľadávanie hracej plochy. Začne od prvej bunky a skúša čísla podobne ako backtracking. Ak číslo vyhovuje, vytvorí kópiu dosky, vloží ju do zásobníka a pokračuje k ďalšej bunke. Keď sa dostane do slepej uličky, vráti sa k predchádzajúcemu stavu.

Výhody: Alternatíva k backtrackingu s jasnejším spracovaním pomocou zásobníka.

Nevýhody: Tiež skúša mnoho možností, takže môže byť pomalý.

Minimálny počet krokov: 108 (Stav 8)

Maximálny počet krokov: 2264 (Stav 7)

Priemerný počet krokov: 671

3. Forward Checking (Kontrola dopredu)

Ako funguje: Algoritmus vytvára zoznam možností pre každú bunku (možné čísla, ktoré môžu byť v bunke). Predtým, než vloží číslo do bunky, kontroluje, či toto vloženie nezruší všetky možnosti pre iné bunky. Ak áno, túto možnosť zamietne.

Výhody: Znižuje počet potrebných pokusov odstránením neplatných možností dopredu (zlepšenie efektivity).

Nevýhody: Vyžaduje viac pamäte a výpočtového času na udržiavanie zoznamov možností.

Na základe poskytnutých údajov môžeme vytvoriť prehľadný štatistický opis pre jednotlivé algoritmy (Backtracking, DFS a Forward Checking) za všetky stavy:

Minimálny počet krokov: 45 (viackrát: Stavy 1, 2, 4, 6, 7, 8, 9, 10)

Maximálny počet krokov: 61 (Stav 5)

Priemerný počet krokov: 47.8

Porovnanie efektívnosti algoritmov

Forward Checking je jednoznačne najefektívnejší algoritmus, pričom jeho počet krokov je konzistentne nízky (s minimálnymi odchýlkami).

DFS (Depth-First Search) dosahuje v priemere výrazne nižší počet krokov než Backtracking, hoci rozdiel medzi maximálnymi hodnotami (2264 vs. 4483) naznačuje vyššiu variabilitu.

Backtracking je najmenej efektívny algoritmus, s najvyšším priemerom a najväčšou variabilitou počtu krokov.

Celkové zhrnutie:

- Forward Checking ukazuje stabilné a konzistentné výkony.
- DFS je vhodnejší ako Backtracking, avšak stále je menej efektívny než Forward Checking.
- Backtracking je najmenej efektívny pre riešenie daných problémov, čo naznačujú jeho vysoké priemery aj maximálne hodnoty.

Stav 1.

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

Štatistiky:

Backtracking: 2071 krokov

DFS: 1058 krokov

Forward Checking: 45 krokov

Stav 2.

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

Štatistiky:

Backtracking: 355 krokov

DFS: 200 krokov

Forward Checking: 45 krokov

Stav 3.

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

Štatistiky:

Backtracking: 731 krokov

DFS: 388 krokov

Forward Checking: 57 krokov

Stav 4.

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

Štatistiky:

Backtracking: 227 krokov

DFS: 136 krokov

Forward Checking: 45 krokov

Stav 5.

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

Štatistiky:

Backtracking: 1549 krokov

DFS: 797 krokov

Forward Checking: 61 krokov

Stav 6.

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

Štatistiky:

Backtracking: 819 krokov

DFS: 432 krokov

Forward Checking: 45 krokov

Stav 7.

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

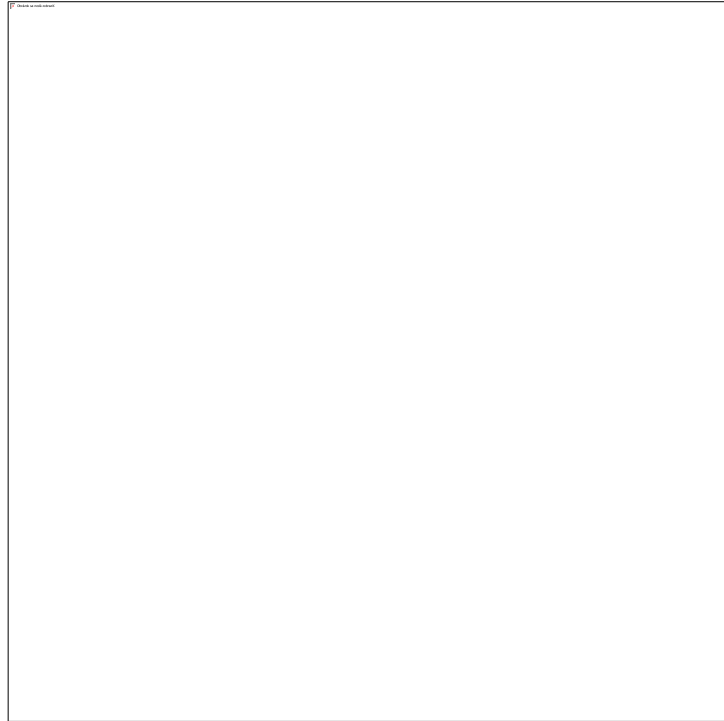
Štatistiky:

Backtracking: 4483 krokov

DFS: 2264 krokov

Forward Checking: 45 krokov

Stav 8.



Štatistiky:

Backtracking: 233 krokov

DFS: 108 krokov

Forward Checking: 45 krokov

Stav 9.

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

Štatistiky:

Backtracking: 267 krokov

DFS: 156 krokov

Forward Checking: 45 krokov

Stav 10.

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

Štatistiky:

Backtracking: 296 krokov

DFS: 171 krokov

Forward Checking: 45 krokov