

Semestrální práce z ALG1

Úloha 24. Redukce matic.

Semin German
Technická Univerzita v Liberci, FM

Specifikace úlohy. Co má program dělat a jak.

Podle původní úlohy v souborů všech semestrálních úloh:

*“Zapište program, který bude provádět redukci čtvercové matice. V načtené matici má program **vyhledat nenulový prvek** pro který platí, že **je jediným nenulovým prvkem na řádku a ve sloupci**. Pokud takový prvek v matici neexistuje, potom má program vypsát informaci, že matici nelze dále upravit. Pokud je prvek nalezen, potom **má program odstranit řádek a sloupec**, ve kterém se nalezený prvek nachází a **opakovaně přejít k vyhledání dalšího prvku v již upravené matici**, který splňuje uvedenou podmínku.”*

Prostě řečeno, co to znamená? Program má načíst čtvercovou matici a najít sloupce a řádky které mají jenom jediný nenulový prvek a je odstranit (jak na obrázku):

$$\begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 1 & 0 & 2 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 3 \\ 1 & 2 \end{bmatrix} \quad \left| \quad \begin{bmatrix} 1 & 0 & 0 & 5 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 7 & 0 & 0 & 1 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 0 & 5 \\ 0 & 1 & 0 \\ 7 & 0 & 1 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 5 \\ 7 & 1 \end{bmatrix}$$

Specifikace úlohy. Předpoklady.

Protože program zpracovává matice a úloha říká o celých číslech, tak předpokládáme že uživatel bude zadávat jenom celé čísla; ale pro lepší *user experience* přidáme kontrolu vstupní hodnot a když uživatel zkusi zadat špatnou hodnotu (reálné číslo nebo nějaký nečíselný řetězec) tak ho o tom upozorňujeme a ukončíme činnost programu aby to nedošlo k výjimce.

Rozmer matice

2

Zadej matici ✓

1 2

3 4

<Program to zpracovává>

Rozmer matice

3

Zadej matici ✗

1 a b

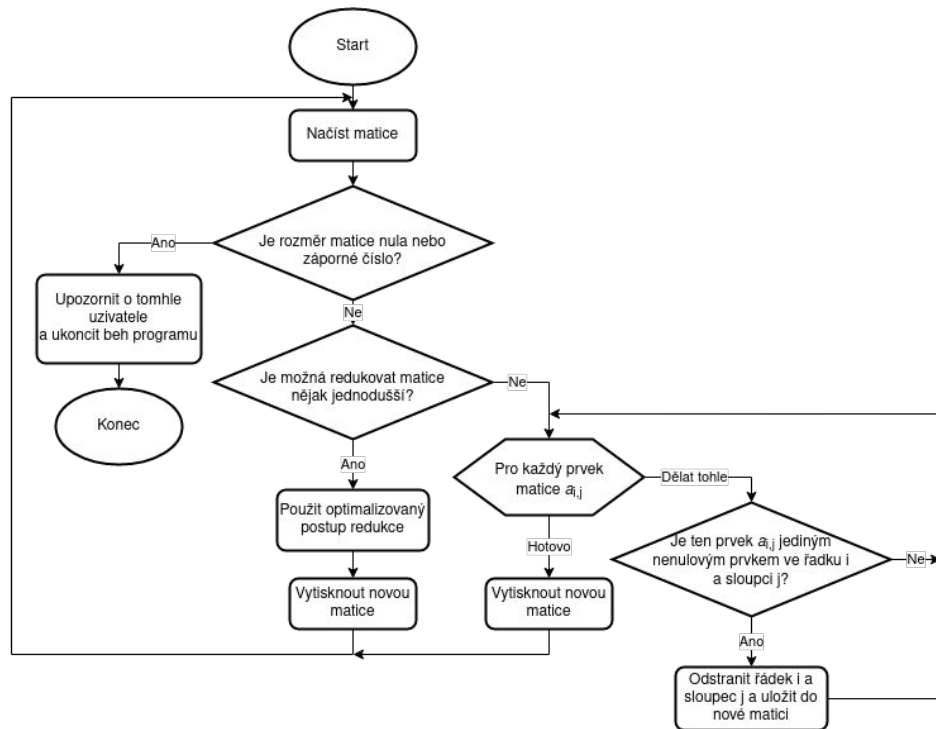
Ocekava se cela hodnota.

Program se ukonci.

Návrh řešení. Jak tuhle úlohu je možná vyřešit.

Tuhle úlohu je možná vyřešit pomocí následujícího algoritmu (koukej doprava):

1. Načíst matice a když je rozměr menší nebo se rovná nule tak o tom vypsát zprávu a ukončit program.
2. Zkusit redukovat matice jednodušším způsobem (např. když matice má tvar 2×2 .)
3. Jinak použít obecný postup redukce, pro každý prvek matice zkontrolovat, jestli tento prvek je jediným nenulovým prvkem matice a v tomto případě tenhle řádek a sloupec odstranit.
4. Vypsát novou matice.
5. Začít běh programu znovu.



Nejzajímavější část kódu. Rychlejší redukce matic 2x2.

Kontroluje když zadana matice je dvoupřvkova matice a může být zredukována do jednoho prvku (když má tvar $\begin{bmatrix} X & 0 \\ 0 & X \end{bmatrix}$ nebo $\begin{bmatrix} 0 & X \\ X & 0 \end{bmatrix}$.)

Params: mat - matice

Returns: -2 když matice nemá tvar 2x2, -1 když nemůže být zredukována žádným způsobem, 0 když může být zredukována a výsledkem bude číslo na pozici $x=1, y=1$, 1 když výsledkem bude číslo na pozici $x=1, y=0$

See Also: REDUCIBLE_SQ2_MATRIX_LC_ZERO,
REDUCIBLE_SQ2_MATRIX_RC_ZERO,
NON_SQ2_MATRIX,
NON_REDUCIBLE_SQ2_MATRIX

5 usages dreamscached

```
129 @
130 public static int isSq2ReducesToSq1Matrix(int[][] mat) {
131     // Matice nemá tvar 2x2.
132     if (mat.length != 2 || mat[0].length != 2) return NON_SQ2_MATRIX;
133     // Matice má nuly na (0,0) a (1,1) - nula je v levém rohu.
134     if (mat[0][0] == 0 && mat[1][1] == 0) return REDUCIBLE_SQ2_MATRIX_LC_ZERO;
135     // Matice má nuly na (0,1) a (1,0) - nula je v pravém rohu.
136     if (mat[0][1] == 0 && mat[1][0] == 0) return REDUCIBLE_SQ2_MATRIX_RC_ZERO;
137     // Jinak nemůže žádným způsobem její redukovat.
138     return NON_REDUCIBLE_SQ2_MATRIX;
139 }
```

Co a proč? Tady je metoda která povolí jednodušší redukovat matice mající rozměry 2x2 a následující tvar:

$$\begin{bmatrix} 0 & x \\ y & 0 \end{bmatrix} \text{ nebo } \begin{bmatrix} x & 0 \\ 0 & y \end{bmatrix}$$

Co to dělá? Tato metoda kontroluje jestli matice má rozměry 2x2 a výše zmíněné tvary a když ano, tak říká když můžeme redukovat a vzít y zleva dolů (tvar zleva) nebo zprava dolů (tvar zprava.)

Dal informace získanou z této metody použijeme v metodě která vlastně redukuje matice a je to rychlejší než pomocí cyklu.

Testování programu. Screenshoty výsledků testů.

Příklad z původní úlohy

```
Rozmer matice
3
Zadej matici
15 0 16
0 23 0
16 0 18
Redukovana matice (2 x 2)
15 16
16 18
```

Příklad s ukončením programu po zadání záporné hodnoty

```
Rozmer matice
-1
Program se ukonci.
```

“Skalár”, matice z jednoho prvku; také příklad z nereducibilní matice

```
Rozmer matice
1
Zadej matici
1337
Redukovana matice (1 x 1)
1337
```

Příklad s žádáním nevhodné hodnoty a ukončením programu

```
Rozmer matice
3
Zadej matici
1 a 3
Ocekava se cela hodnota. Program se ukonci.
```

Process finished with exit code 1

Matice 4x4, mající dva řádky a sloupce s jednou nenulovou hodnotou

```
Rozmer matice
4
Zadej matici
1 0 0 5
0 1 0 0
0 0 1 0
7 0 0 1
Redukovana matice (2 x 2)
1 5
7 1
```

Konec prezentace

Je to všechno, co mám pro dnešní hodinu. Děkuju všem za pozornost!

Tato prezentace a celý zdrojový kód
jsou uložené v mém Git repozitáře:

bit.ly/3VifiQq

