



TECHNICKÁ UNIVERZITA V LIBERCI  
Fakulta mechatroniky, informatiky  
a mezioborových studií ■

# Transformace zrcadlení matic

**Semestrální práce**

Předmět: MTI/ALG1 – Algoritmizace a programování 1  
Studijní obor: B0613A140005 - Informační technologie  
Autor práce: Tomáš Kulhavý  
Vedoucí práce: Ing. Jana Vítvarová, Ph.D.

### Obsah

1. Specifikace požadavků.....	3
2. Návrh řešení.....	4
1 Zrcadlení podle hlavní diagonály.....	4
2 Zrcadlení podle vedlejší diagonály.....	4
3 Zrcadlení podle horizontální osy.....	5
4 Zrcadlení podle vertikální osy.....	5
3. Protokol z testování.....	6
Přílohy.....	9

# 1. Specifikace požadavků

Program by měl umět načíst dvě čtvercové matice stejné velikosti a měl by zjistit zda druhá matice vznikne jednoduchou transformací zrcadlení podle hlavní, vedlejší diagonály nebo horizontální, vertikální osy matice.

Program má umožnit při jednom spuštění zpracování libovolného počtu zadání. Pro každou úlohu program načte celé číslo jako rozměr zpracovávaných matic, poté budou načítány jednotlivé matice po řádcích. Program vyzve uživatele k zadání celého čísla. Všechny vstupy až na zadávání čísel do matice jsou ošetřeny, aby bylo možné napsat pouze celé číslo.

Pro každé zadání program vypíše informaci, zda druhá ze zadaných matic vznikne z první některou z uvedených transformací. Program výsledek svého šetření vypisuje do konzole jako jednu z následujících krátkých textových zpráv: „Zrcadlení dle hlavní diagonály“, „Zrcadlení dle vedlejší diagonály“, „Zrcadlení dle horizontální osy“, „Zrcadlení dle vedlejší osy“ nebo „Nenalezena transformace“. Po načtení záporného nebo nulového čísla místo rozměru matice program skončí svoji činnost.

Pro dvě zadané matice může program najít i více transformací, které vypíše na výstup do konzole.

## 2. Návrh řešení

Program je rozdělen na hlavní třídu Main.java. Ta slouží k rozcestníku programu. Uživateli vypíše do konzole menu pomocí cyklu do-while. Uživatel má na výběr vánoční úlohu, semestrální práci nebo možnost z programu odejít.

Dále program obsahuje třídu MatrixUI.java. Ta komunikuje s uživatelem a slouží k načítání rozměru matice a k plnění samotné matice. Rozměry matice a prvky v ní musí být pouze celá čísla. Matice je plněna pomocí cyklu for, která ji plní postupně po řádcích.

K vyhodnocení transformace matice je zde třída MatrixTools.java. Každá metoda přijímá dvě matice celých čísel a vracejí boolean. Pokud je matice symetrická podle daných požadavků vrátí „true“, jestliže ne tak vrátí „false“.

### 1 Zrcadlení podle hlavní diagonály

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

Obrázek 1: Zrcadlení podle hlavní diagonály (porovnávání členů)

Algoritmus pro zjištění transformace zrcadlení podle hlavní diagonály funguje přes cykly for. Postupně z načtených matic si vybere řádky např. u matice 3x3 druhý řádek z první matice a první řádek z druhé matice. A pomocí druhého cyklu for si vybere první číslo z druhého řádku první matice a druhé číslo z prvního řádku druhé matice. Pokud se tyto prvky rovnají, metoda pokračuje dál a na konci vrátí „true“. Jestliže se tyto čísla nerovnají metoda vrátí „false“ a metoda se ukončí.

### 2 Zrcadlení podle vedlejší diagonály

Algoritmus pro zjištění transformace zrcadlení podle vedlejší diagonály funguje podobně jako algoritmus podle hlavní diagonály. Porovnává první řádek první matice a poslední řádek druhé matice. Z první matice vybere první číslo a z druhé matice vybere poslední číslo.

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

Obrázek 2: Zrcadlení podle vedlejší diagonály (porovnávání členů)

### 3 Zrcadlení podle horizontální osy

Algoritmus pro zjištění transformace zrcadlení podle horizontální osy také funguje přes cykly for. Postupně z načtených matic si vybere řádky např. u matice 3x3 první řádek z první matice a poslední řádek z druhé matice. A pomocí druhého cyklu for si vybere první číslo z prvního řádku první matice a první číslo z posledního řádku druhé matice. Pokud se tyto prvky rovnají, metoda pokračuje dál a na konci vrátí „true“. Jestliže se tyto čísla nerovnají metoda vrátí „false“ a metoda se ukončí.

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

Obrázek 3: Zrcadlení podle horizontální osy (porovnávání členů)

### 4 Zrcadlení podle vertikální osy

Algoritmus pro zjištění transformace zrcadlení podle vertikální osy také funguje podobně jako algoritmus podle horizontální osy. Ale porovnává první sloupec první matice a poslední sloupec druhé matice. Z první a druhé matice vybere první číslo ze sloupců a porovná je. Pokud se rovnají vrátí „true“, pokud ne tak vrátí „false“.

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

Obrázek 4: Zrcadlení podle vertikální osy (porovnávání členů)

### 3. Protokol z testování

Číslo testu	Typ testu	Očekávaný výsledek	Skutečný výsledek	Prošel
1	běžná hodnota	Všechny možné stavy	Všechny možné stavy	ano

```

1  1  1
1  1  1
1  1  1

1  1  1
1  1  1
1  1  1

Zrcadlení dle horizontální osy.
Zrcadlení dle vertikální osy.
Zrcadlení dle hlavní diagonály.
Zrcadlení dle vedlejší diagonály.

```

Obrázek 5: Test č. 1 – Výpis všech možných stavů

Číslo testu	Typ testu	Očekávaný výsledek	Skutečný výsledek	Prošel
2	běžná hodnota	Nenalezena transformace	Nenalezena transformace	ano

```

1  2  3
4  5  6
7  8  9

9  8  7
6  5  4
3  2  1

Nenalezena transformace.

```

Obrázek 6: Test č. 2 - Žádná transformace

## Transformace zrcadlení matic

Číslo testu	Typ testu	Očekávaný výsledek	Skutečný výsledek	Prošel
3	běžná hodnota	Výstup z úlohy matic	Výstup z úlohy matic	ano

```

1 - Vánoční úloha
2 - Semestrální práce
0 - EXIT
2

Rozměr matic: -1

1 - Vánoční úloha
2 - Semestrální práce
0 - EXIT

```

Obrázek 7: Test č. 3 -  
Výstup z úlohy matic

Číslo testu	Typ testu	Očekávaný výsledek	Skutečný výsledek	Prošel
4	běžná hodnota	zrcadlení dle vertikální osy	zrcadlení dle vertikální osy	ano

```

1  2  3
4  5  6
7  8  9

7  8  9
4  5  6
1  2  3

Zrcadlení dle vertikální osy.

```

Obrázek 8: Test č. 4 - Zrcadlení  
dle vertikální osy

## Transformace zrcadlení matic

Číslo testu	Typ testu	Očekávaný výsledek	Skutečný výsledek	Prošel
5	Nevalidní vstup	chyba	chyba	ano

Obrázek 9: Test č. 5 - Nevalidní vstup

První matice:

1.3 4 5 6

Exception in thread "main" java.util.InputMismatchException Create breakpoint

at java.base/java.util.Scanner.throwFor(Scanner.java:939)

at java.base/java.util.Scanner.next(Scanner.java:1594)

at java.base/java.util.Scanner.nextInt(Scanner.java:2258)

at java.base/java.util.Scanner.nextInt(Scanner.java:2212)



# Přílohy

## Seznam obrázků

Obrázek 1: Zrcadlení podle hlavní diagonály (porovnávání členů).....	4
Obrázek 2: Zrcadlení podle vedlejší diagonály (porovnávání členů).....	4
Obrázek 3: Zrcadlení podle horizontální osy (porovnávání členů).....	5
Obrázek 4: Zrcadlení podle vertikální osy (porovnávání členů).....	5
Obrázek 5: Test č. 1 – Výpis všech možných stavů.....	6
Obrázek 6: Test č. 2 - Žádná transformace.....	6
Obrázek 7: Test č. 3 - Výstup z úlohy matic.....	7
Obrázek 8: Test č. 4 - Zrcadlení dle vertikální osy.....	7
Obrázek 9: Test č. 5 - Nevalidní vstup.....	8