

Semestrální práce – ALG1

Jakub Dvořák 20.12.22

Úloha číslo 12

Zadání

Zapište **program pro testování generátoru náhodných čísel**. Generátor náhodných čísel by měl generovat zcela náhodnou posloupnost hodnot, kde aktuálně generovaná hodnota není závislá na hodnotách předchozích. Z náhodně generované posloupnosti hodnot v rozsahu od a do b vytvoříme dvojice čísel (x_i, y_i) , kde první z hodnot x_i bude náhodně generované číslo v i -tém kroku a druhé z čísel y_i bude náhodně generované číslo v $(i+k)$ -tém kroku. Pokud generujeme celkem n čísel potom takovýchto dvojic z generované řady lze vytvořit celkem $n-k$. Pro výslednou sadu dvojic můžeme vyhodnotit korelační koeficient. Pro dostatečnou sadu hodnot a pro $k \neq 0$ by měl mít vypočtený korelační koeficient hodnotu blízkou 0. Zapište program, který bude pro zadané hodnoty parametrů n a k vyhodnocovat korelační koeficient.

Korelační koeficient (Pearsonův korelační koeficient) r vypočtete dle vzorce:

$$r = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}}$$

Poznámka: korelační koeficient odhalí pouze případnou lineární závislost.

Specifikace vstupu

Program má umožnit při jednom spuštění zpracování libovolného počtu zadání. Před zadáváním další úlohy nechť program vypíše dotaz uživateli, zda pokračovat ve zpracování či nikoli – odpověď uživatele bude znak 'a' nebo 'n' (zadáno malým nebo velkým písmenem). Program má skončit v případě, že odpověď uživatele je 'n'. Při načítání vstupních hodnot program načte nejdříve meze a a b intervalu náhodně generovaných čísel a poté dvě celá kladná čísla n a k .

Program bude fungovat za předpokladu, že uživatel zadá n (počet náhodných čísel), jako celé, nezáporné číslo větší než 0 a k (krok mezi dvojicemi).

Návrh řešení

1. Načíst minimum (*min*), maximum(*max*), počet(*n*) a krok(*k*) náhodných hodnot od uživatele
2. Zjistit jestli platí, že $\text{min} < \text{max}$ a $0 < n > k$
3. Vygenerovat posloupnost(*P*) náhodných čísel o *n* prvcích, pomocí „*Math.random()* * (*max* – *min* + 0.01) + *min*“
4. Vytvořit matici(*M*) o *n-k* řádcích a 2 sloupcích
5. Naplnit matici *M* prvky posloupnosti *P*, kde *M*[*i*][0] je *P*[*i*] a *M*[*i*][1] je *P*[*i+k*]
6. Vypočítat průměrnou hodnotu sloupce 0 (*avrgLine0*) a 1 (*avrgLine1*) matice *M*, pomocí samostatné metody
7. Vypočítat sumu (*M*[*i*][0] – *avrgLine0*)*(*M*[*i*][1] – *avrgLine1*), sumu (*M*[*i*][0] – *avrgLine0*)² a sumu (*M*[*i*][1] – *avrgLine1*)², kde *i* bude nabývat hodnot od 0 do *M.length*
8. Vydělit a odmocnit podle Pearsonova vzorce získané hodnoty
9. Získaný korelační koeficient vypsát

Protokol z testování

Číslo testu	Typ testu	Očekávaný výstup	Skutečný výstup	Prošel (ano/ne)
1	Běžný 2,37 ; 5,73 7,41 ; 4,62 5,73 ; 9,9 4,62 ; 4,03	0,058	0,058	ano
2	Limitní Zadání mnohem větších hodnot	Přetížení kódu	error	ne
3	Limitní Min=max	Text + dělení nulou	Text + NaN	ano
4	Nevalidní Záporné k	Tex + error	Text + error	ano
5	k>n	Text + error	Text + error	ano

```

Chcete pokračovat ve zpracování? (a/n)
a
Zadejte:
Dolní mez intervalu:
1
Horní mez intervalu:
10
Počet generovaných hodnot:
6
Delku kroku pro vytváření dvojic:
2
2,37 5,73
7,41 4,62
5,73 9,90
4,62 4,03
Vypočtený korelační koeficient má hodnotu: 0,058

```

D4		: <input type="checkbox"/> <input checked="" type="checkbox"/> f_x		=CORREL(A1:A4;B1:B4)		
	A	B	C	D	E	F
	2,37	5,73				
	7,41	4,62				
	5,73	9,9				
	4,62	4,03		0,057809		

2.

```

Chcete pokračovat ve zpracování? (a/n)
a
Zadejte:
Dolní mez intervalu:
1
Horní mez intervalu:
1000000000
Počet generovaných hodnot:
500000000
Delku kroku pro vytváření dvojic:
45
Exception in thread "main" java.lang.OutOfMemoryError: Java heap space
    at DvořákJSemestrálProject.DvořákJSemestrálProject.getRandomArray(DvořákJSemestrálProject.java:64)
    at DvořákJSemestrálProject.DvořákJSemestrálProject.getKorelKoef(DvořákJSemestrálProject.java:37)
    at DvořákJSemestrálProject.UI.semestrálníPráce(UI.java:86)
    at DvořákJSemestrálProject.UI.main(UI.java:28)
C:\Users\kubad\AppData\Local\NetBeans\Cache\8.2rc\executor-snippets\run.xml:53: Java returned: 1
BUILD FAILED (total time: 28 seconds)

```

3.

```
Chcete pokračovat ve zpracovani? (a/n)
a
Zadejte:
Dolní mez intervalu:
-100
Horní mez intervalu:
100
Počet generovaných hodnot:
-100
Delku kroku pro vytváření dvojic:
10
Exception in thread "main" java.lang.NegativeArraySizeException
    at DvořákJSemestrálProject.DvorakJSemestrálProject.getRandomArray(
    at DvořákJSemestrálProject.DvorakJSemestrálProject.getKorelKoef(Dv
    at DvořákJSemestrálProject.UI.semestrálníPrace(UI.java:86)
    at DvořákJSemestrálProject.UI.main(UI.java:28)
C:\Users\kubad\AppData\Local\NetBeans\Cache\8.2rc\executor-snippets\run.xml
BUILD FAILED (total time: 1 minute 19 seconds)
```

4.

```
Chcete pokračovat ve zpracovani? (a/n)
a
Zadejte:
Dolní mez intervalu:
0
Horní mez intervalu:
1000
Počet generovaných hodnot:
1000
Delku kroku pro vytváření dvojic:
-2
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: -2
    at DvořákJSemestrálProject.DvorakJSemestrálProject.getPairArray(DvorakJSemestrálProject.java:86)
    at DvořákJSemestrálProject.DvorakJSemestrálProject.getKorelKoef(DvorakJSemestrálProject.java:38)
    at DvořákJSemestrálProject.UI.semestrálníPrace(UI.java:86)
    at DvořákJSemestrálProject.UI.main(UI.java:28)
C:\Users\kubad\AppData\Local\NetBeans\Cache\8.2rc\executor-snippets\run.xml:53: Java returned: 1
BUILD FAILED (total time: 18 seconds)
```

5.

```
Zadejte:
Dolni mez intervalu:
0
Horni mez intervalu:
1000
Pocet generovanych hodnot:
1000
Delku kroku pro vytvoreni dvojic:
1001
Exception in thread "main" java.lang.NegativeArraySizeException
    at DvořákJSemestralProject.DvorakJSemestralProject.getPairArray(DvorakJSemestralProject.java:82)
    at DvořákJSemestralProject.DvorakJSemestralProject.getKorelKoef(DvorakJSemestralProject.java:38)
    at DvořákJSemestralProject.UI.semestralniPrace(UI.java:86)
    at DvořákJSemestralProject.UI.main(UI.java:28)
C:\Users\kubad\AppData\Local\NetBeans\Cache\8.2rc\executor-snippets\run.xml:53: Java returned: 1
BUILD FAILED (total time: 24 seconds)
```