



ZÁPADOČESKÁ
UNIVERZITA
V PLZNI

FAV *Fakulta
aplikovaných
věd*

Výkonnost a spolehlivost číslicových systémů

**Okruh 1: Generování náhodných čísel
Zadání: 1 – Lichoběžníkové rozdělení**

Jméno a příjmení: Jan Tichava
Osobní číslo: A07103
E-mail: jtichava@students.zcu.cz
Narozen: 23. březen 1984

1. Zadání

Vytvořte generátor rozdělení jako funkci v jazyce Java, C či Pascal/Delphi (parametry rozdělení jsou zároveň parametry funkce generátoru) s využitím vhodné metody (inverzní transformace, kompoziční, vylučovací, atd.),

Lichoběžníkové rozdělení

- $f(x) = 0$ pro $x < 0$
- $f(x)$ lin. roste pro $a > x > 0$
- $f(x)$ je konst pro $b > x > a$
- $f(x)$ lin. klesá pro $c > x > b$
- $f(x) = 0$ pro $x > c$

Poznámka: U tohoto příkladu není pro dosažení plného počtu bodů nutné počítat teoretické D., Pokud jej spočítáte, dostanete bod navíc.

2. Generátor

Odvození E teorie

Výpočet střední hodnoty při spojitém rozdělení:

$$EX = \int_R xf(x)dx$$

Zdroj: http://cs.wikipedia.org/wiki/Střední_hodnota

Střední hodnotu počítám po částech pomocí tří integrálů, vlastní výpočet integrálů byl symbolicky proveden na kalkulačce TI-89. Výsledek jednotlivých integrálů se sečte a výsledkem je střední hodnota.

$$\int_0^a x \cdot \left(\frac{h}{a}x\right) dx = \frac{a^2 \cdot h}{3}$$

$$\int_a^b x \cdot h dx = -\frac{(a^2 - b^2) \cdot h}{2}$$

$$\int_b^c x \cdot \left(h - \frac{h}{c-b} \cdot (x-b)\right) dx = -\frac{(5 \cdot b^2 - b \cdot c - 4 \cdot c^2) \cdot h}{6}$$

Poznámka: Ilustrační obrázek viz *Popis generátoru*

Odvození D _teorie

Výpočet rozptylu při spojitém rozdělení:

$$\sigma^2 = \int_{-\infty}^{\infty} [x - EX]^2 f(x) dx$$

Zdroj: [http://cs.wikipedia.org/wiki/Rozptyl_\(statistika\)](http://cs.wikipedia.org/wiki/Rozptyl_(statistika))

Uvádím pouze tento teoretický vztah, v programu není teoretický rozptyl počítán, protože není vyžadován.

Výpočet E _vypocet

Skutečnou střední hodnotu počítám dle vzorce (skripta):

$$E\{x\} \doteq \bar{x} = \frac{1}{N} \sum_{i=1}^N x_i$$

Hodnoty x_i průběžně sčítám během generování, na konci součet vydělím počtem čísel.

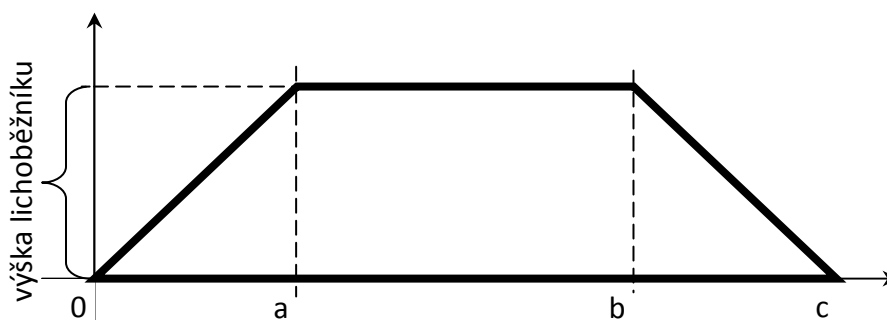
Výpočet D _vypocet

Rozptyl počítám celý na konci, po dokončení generování dle vzorce (skripta):

$$D\{x\} \doteq \frac{1}{N-1} \sum_{i=1}^N (\bar{x} - x_i)^2$$

Hodnotu \bar{x} použiji z výpočtu výše.

Popis generátoru



Pro generování náhodných čísel využívám vylučovací metodu:

- vygenerovat náhodné číslo $x \in \langle 0; c \rangle$
- vygenerovat náhodné číslo z intervalu $y \in \langle 0; \text{výška lichoběžníku} \rangle$

- x a y považujeme za souřadnice v prostoru
- náhodné číslo patří do lichoběžníkového rozdělení, pokud platí $y < f(x)$, jinak postup opakujeme

Více viz dokumentace JavaDoc nebo zdrojový kód.

Histogram

Při generování histogramu se interval rozdělí na 30 stejně velkých intervalů. Každé číslo se zařadí do jednoho z intervalů. Počet hvězdiček na řádku se odvíjí od maximálního počtu hvězdiček na řádek. Jako maximum jsem zvolil 50, pro případ, že by někdo náhodou používal terminál s 80 znaky na řádek. Na konec řádky se dopíše informativně hodnota čísla a ještě počet čísel, které spadají do daného intervalu.

Statistiky

Na závěr se vypíše základní statistiky:

- jak dlouho se generovalo zadané množství čísel s lichoběžníkovým rozdělením
- rychlost generování čísel s lichoběžníkovým rozdělením za sekundu
- skutečný počet generovaných náhodných čísel
- rychlost generování náhodných čísel za sekundu

Pokud je čas příliš krátký nebo generovaná malá množina čísel, nemusí být rychlosti generování přesné.

Ostatní

Soubor VSP.jar obsahuje:

- přeložené CLASS soubory
- zdrojové JAVA soubory
- dokumentaci JavaDoc

Maximální počet čísel, která je možné generovat je něco přes 2 000 000, jinak nutné zvětšit Javě heap space, aby nedocházelo k výjimce `OutOfMemoryError`. Vzhledem k tomu, že v zadání se píše maximálně o statisících, tak jsem `-Xmx` parametr nepřidával do `run.bat`.

3. Závěr

Generování náhodných čísel s lichoběžníkovým rozdělením pomocí vylučovací metody je poměrně jednoduchá metoda a naprogramování nezabralo příliš času. Pouze v seznamu zadání jsem si nevšiml, že první zadání má číslo 0, takže jsem nejprve napsal generátor náhodného rozdělení :-).

4. Výstup programu

```
E_teorie=12.917  
E_vypocet=12.832  
D_vypocet=43.745
```

```
***** 1.133 (count: 55)  
***** 2.2 (count: 187)  
***** 3.267 (count: 306)  
***** 4.333 (count: 419)  
***** 5.4 (count: 511)  
***** 6.467 (count: 585)  
***** 7.533 (count: 511)  
***** 8.6 (count: 546)  
***** 9.667 (count: 555)  
***** 10.733 (count: 503)  
***** 11.8 (count: 516)  
***** 12.867 (count: 562)  
***** 13.933 (count: 498)  
***** 15.0 (count: 546)  
***** 16.067 (count: 528)  
***** 17.133 (count: 458)  
***** 18.2 (count: 449)  
***** 19.267 (count: 369)  
***** 20.333 (count: 335)  
***** 21.4 (count: 321)  
***** 22.467 (count: 304)  
***** 23.533 (count: 246)  
***** 24.6 (count: 203)  
***** 25.667 (count: 177)  
***** 26.733 (count: 127)  
***** 27.8 (count: 107)  
***** 28.867 (count: 53)  GB  
** 29.933 (count: 23)
```

```
10000 generated in 13 ms  
Speed is 769231 numbers per second  
Need to generate 15008 numbers  
Real computer generator speed: 1154462 numbers per second  
Note: Statistics may not be accurate for short times and for small set of numbers  
=====
```

```
E_teorie=1800.0  
E_vypocet=1799.259  
D_vypocet=727222.686
```

```
*** 132.4 (count: 3553)  
***** 265.8 (count: 10591)  
***** 399.2 (count: 17911)  
***** 532.6 (count: 24840)  
***** 666.0 (count: 32129)  
***** 799.4 (count: 39284)  
***** 932.8 (count: 46449)  
***** 1066.2 (count: 52288)  
***** 1199.6 (count: 53427)  
***** 1333.0 (count: 53464)  
***** 1466.4 (count: 52952)  
***** 1599.8 (count: 53201)  
***** 1733.2 (count: 53048)  
***** 1866.6 (count: 53233)  
***** 2000.0 (count: 53801)  
***** 2133.4 (count: 51770)  
***** 2266.8 (count: 47854)  
***** 2400.2 (count: 44597)  
***** 2533.6 (count: 40859)  
***** 2667.0 (count: 37250)  
***** 2800.4 (count: 33631)
```

```
***** 2933.8 (count: 30249)
***** 3067.2 (count: 26692)
***** 3200.6 (count: 23189)
***** 3334.0 (count: 19471)
***** 3467.4 (count: 15852)
***** 3600.8 (count: 12451)
***** 3734.2 (count: 8906)
***** 3867.6 (count: 5323)
** 4001.0 (count: 1735)
```

1000000 generated in 1022 ms
Speed is 978474 numbers per second
Need to generate 1601137 numbers
Real computer generator speed: 1566670 numbers per second
Note: Statistics may not be accurate for short times and for small set of numbers

qB