

Języki programowania, Środa 11.12.2014, 16:15-7:45

do generowania liczb pseudolosowych z trzech różnych rozkładów (jednorodnego, Gaussa oraz Poissona).

Klasa do generowania liczb losowych .

Klasa będzie się nazywać `Random`. Jej plik nagłówkowy powinien zawierać:

- pola `double fMin` i `double fMax` – przedział, z którego będą losowane liczby
- konstruktor domyślny – ustawia ziarno z zegara systemowego: `srand(time(NULL))` oraz `fMin` na 0 i `fMax` na 1
- Konstruktor z parametrami ustawiający pola `fMin` i `fMax`; `srand` ustawiamy analogicznie jak w konstruktorze bez parametrów.
- metoda `double Exec()` - metoda wirtualna, zwraca liczbę pseudolosową z rozkładu jednorodnego z przedziału `<fMin;fMax`)

Następnie tworzymy klasę `Gauss`, która dziedziczy z klasy `Random`. Powinna ona zawierać:

- pola typu `double fMean`, `double fSigma`, które przechowują parametry rozkładu Gaussa: średnią i odchylenie standardowe.
- konstruktor z parametrami ustawiający wszystkie pola
- metodę `double Exec()` - zwracającą liczbę pseudolosową z rozkładu Gaussa o zadanej średniej i odchyleniu standardowym (szczegóły implementacji patrz **Uwaga1!**)

Zestaw bibliotek niezbędnych do stworzenia generatora:

`cstdlib`, `ctime`, `cmath`

Należy użyć funkcji `rand()` - zwracającej liczbę całkowitą pseudolosową z rozkładu `<0;RAND_MAX`), gdzie `RAND_MAX` jest wielką liczbą - stałą zdefiniowaną w bibliotece standardowej.

Piszemy program testujący działanie dwóch wyżej wymienionych klas – 2.5 p.

Tworzymy w funkcji `main` obiekt klasy `Random` (np. `rand1(-2, 3)`;) z ustawionymi wartościami od -2 do 3. Losujemy kilka liczb pseudolosowych i wypisujemy na ekran.

Tworzymy w funkcji `main` obiekt klasy `Gauss` (np. `gaus1(100, 5)`;) z ustawioną średnią na 100 i odchyleniem 5. Losujemy kilka liczb pseudolosowych.

Tworzymy klasę abstrakcyjną (ATD – abstrakcyjny typ danych) – 1 p.

Tworzymy klasę `Uniform`, do której „przerzucamy” z klasy `Random` pola `fMin`, `fMax` i algorytm metody `Exec()` losujący liczbę pseudolosową z rozkładu jednorodnego (powinna posiadać konstruktor z parametrami ustawiającymi `fMin` i `fMax`). Klasa dziedziczy z klasy `Random`.

W klasie `Random` tworzymy teraz metodę `Exec()` jako czysto wirtualną, tzn w pliku `Random.h`:

```
virtual double Exec() = 0;
```

Poprawiamy następnie program główny w taki sposób, by się kompilował.

Wskazówka: wszystkie metody czysto wirtualne muszą istnieć w klasach pochodnych.

Przypomnienie: klasa, która ma co najmniej jedną funkcję czysto wirtualną nazywamy **klasą abstrakcyjną**.

Obsługa plików tekstowych – 1 p.

W programie zapisujemy kilka wylosowanych liczb z rozkładu Gaussa do pliku tekstowego.

Następnie wczytujemy zapisany plik i wypisujemy wszystkie liczby na ekran.

Klasa do generowania liczb z rozkładu Poissona – 0.5 p. należy dodać klasę `Poisson` (czytaj **Uwaga2!**), zachowującą się podobnie do poprzedniej, wywołać na niej metodę `Exec()`.

Uwaga 1!

Algorytm do generowania liczb pseudolosowych z rozkładu Gaussa (metoda Box-Muller'a):

- U, V – dane liczby pseudolosowe z rozkładu jednorodnego $<0;1$)
- Liczba losowa z rozkładu Gaussa o średniej μ i odchyleniu standardowym σ dana jest wtedy wzorem:
$$x = \mu + \sigma \sqrt{-2 \ln(U)} \cos(2\pi V)$$

Uwaga 2!

Algorytm do generowania liczb pseudolosowych z rozkładu Poissona o średniej λ :

- x_0, x_1, \dots - ciąg liczb pseudolosowych z rozkładu jednorodnego
- Jeżeli $x_0 x_1 \dots x_k < e^{-\lambda}$, to k jest liczbą z rozkładu Poissona.