

## Lista 10

### Zadanie 1 (7 pkt)

Zaimplementuj generator liczb pseudolosowych ( $N$ -cyfrowych) korzystając z *Middle Square Method*:

- $N$  podawane jako argument linii komend (*sys.argv*)
- program musi protestować, jeśli  $N$  nie jest liczbą parzystą
- $n$  - liczba punktów do wygenerowania podawane jako argument linii komend
- program musi protestować, jeśli  $n$  nie jest liczbą naturalną
- (opcjonalnie) *seed* podawane jako argument linii komend
- program musi protestować, jeśli podany *seed* nie jest  $N$ -cyfrowy
- jeśli *seed* nie zostanie podany, to powinien być inicjalizowany na podstawie czasu, np. MMSS dla  $N = 4$  lub YYYYMMDDHHMMSS dla  $N = 14$
- wykorzystaj wyjątki, aby wychwycić podanie nieprawidłowych danych
- program powinien generować  $n$  liczb pseudolosowych ( $N$ -cyfrowych) dla zadanego *seed* i przedstawić je na wykresie (podobnie jak na wykładzie link, w szczególności dla  $n = 100$ ,  $N = 4$  i *seed* = 1234 wynik z wykładu powinien zostać odtworzony)

### Zadanie 2 (8 pkt)

Niech ciastko będzie zadane przez rozmiar  $S$  (będący liczbą naturalną).

Niech dziecko będzie zdefiniowane przez łakomstwo  $L$  (będące liczbą naturalną), które definiuje minimalny rozmiar ciastka, które zadowoli dziecko (czyli dziecko przyjmie ciastko, gdy  $S \geq L$ ).

Mając dane  $N$  ciastek o różnych rozmiarach i  $M$  dzieci o różnych łakomstwach, zmaksymalizuj liczbę zadowolonych dzieci. *Jedno dziecko może dostać co najwyżej jedno ciastko!*

Przykład:

Ciastka: [1, 2, 3, 2, 2]

Dzieci: [1, 1, 4, 2, 3, 3]

Maksymalna liczba zadowolonych dzieci: 4

1 dziecko o  $L=4$  -> brak odpowiedniego ciastka

2 dzieci o  $L=3$  -> tylko jedno ciastko o  $S=3$  (zostają [1, 2, 2, 2])

1 dziecko o  $L=2$  -> zostają [1, 2, 2]

2 dzieci o  $L=1$  -> dostają po ciastku i dwa zostają nie użyte

Niech  $S$  i  $L$  przyjmują wartości o 1 do 10

- Napisz skrypt do generowania zestawów danych. Program powinien zawierać:

- funkcję, która przyjmuje dwa argumenty (liczbę dzieci  $M$  i liczbę ciastek  $N$ ) i zwraca losowe zestawy  $S$  i  $L$ ; funkcja wywołana bez podania argumentów powinna użyć losowych wartości  $M$  i  $N$

- funkcję, która przyjmuje trzy argumenty (dwie listy/krotki i ścieżka do pliku) i dopisuje (w nowej linii) listy/krotki do pliku w formacie

C1, C2, C3, ..., CN; L1, L2, L3, ... LM

- funkcję *main*, która pobiera z linii komend dwa argumenty: liczbę zestawów oraz ścieżkę do pliku i wykorzystuje powyższe funkcje, aby utworzyć plik z zestawami danych, np:

```
1, 3, 2, 4; 1, 1, 2, 2
8, 4, 5, 2, 4; 3, 4, 7, 8
...
```

- Napisz moduł do rozwiązywania wyżej opisanego problemu, który zawiera:

- funkcję, która przyjmuje dwa argumenty (listę ciastek i listę dzieci) i zwraca maksymalną liczbę zadowolonych dzieci

- funkcję *main*, która testuje działanie głównej funkcji z pierwszego podpunktu na kilku przykładach

- Napisz program do analizy wygenerowanych zestawów. Program powinien czytać dane wygenerowane skryptem z pierwszego punktu i wykorzystywać moduł napisany w drugim punkcie. Przykładowy wynik działania programu:

```
Ciastka: [1, 2, 3, 2, 2]
Dzieci: [1, 1, 4, 2, 3, 3]
```

Maksymalna liczba zadowolonych dzieci: 4

```
Ciastka: [1, 1, 1]
Dzieci: [2, 3, 1]
```

Maksymalna liczba zadowolonych dzieci: 1