

Kurs języka Python

Lista 4.

Zadanie 1. Zaimplementuj to samo zadanie które wykonałeś/wykonałaś z listy 3., ale wykorzystaj do tego iteratory. Zbadaj, która teraz wersja jest najszybsza.

Przetestuj działanie implementacji dla różnych argumentów, np. dla 10, 100, 1000 etc. Wypisz na konsolę czasy działania dla poszczególnych danych i implementacji w ładnie sformatowany sposób, na przykład

	funkcyjna	skladana	iterator
10	0.001	0.002	0.003
100	0.010	0.020	0.030
1000	0.100	0.200	0.300

Można też plik sformatować tak, aby wynik mógł być wejściem dla jakiegoś pakietu/programu rysującego wykresy, np. *gnuplot* czy *matplotlib*.

Zadanie 2. Zaprogramuj jako iterator kolekcję do przetwarzania plików tekstowych w formie 'akapit po akapicie'. Na przykład aby można było łatwo sformatować plik do postaci html'a:

```
for akapit in kolekcja:
    print('<p>' + akapit + '</p>')
```

Kolekcja powinna korzystać z dowolnego strumienia danych.

Zaprogramuj funkcję `formatuj_akapit(akapit, szerokość)` która zwróci ten sam tekst, jednak żaden wiersz nie będzie dłuższy niż `szerokość` (o ile będzie to możliwe). Jako prezentację działania programu utwórz kolekcję sformatowanych akapitów, np. korzystając z `map`.

Zadanie 3. Zaprogramuj iterator, który przetwarza strumień tekstowy i zwraca kolejne zdania (stringi) z tego strumienia. Zaprogramuj funkcję `korekta(zdanie)` która sprawdza, czy zdanie zaczyna się wielką literą i kończy się kropką, oraz czy po kropce jest jakiś biały znak, a jako wynik zwraca skorygowane zdanie.

Jako prezentację działania programu utwórz kolekcję skorygowanych zdań, np. korzystając z `map` (w wersji 3.*) lub analogicznej funkcji z pakietu `itertools`.

Zadanie 4. Zaprogramuj iterator który przetwarza strumień tekstowy i zwraca kolejne słowa z tekstu (dla utrudnienia uwzględnij dzielenie słów na końcach wierszy), pomijając białe znaki i znaki interpunkcyjne. Korzystając z tej implementacji zaprogramuj obliczanie statystyki długości słów w tekście, tj. ile jest słów długości 1, ile długości 2 etc.

Na kolejną pracownię zaprogramuj zadanie 1. oraz jedno z zadań 2–4. Zadanie 1. jest warte 2 pkt., zadania 2–4: 3 pkt. Do zdobycia jest więc 5 pkt.

Marcin Młotkowski