

Algorytmy i struktury danych

Stos. Dodatkowe zadania.

Zad. 1 Napisz program którego zadaniem jest obliczenie sumy dwóch bardzo dużych liczb naturalnych (np. zawierających więcej niż 60 znaków, nieograniczonych).

- Zdefiniuj klasę Addition, która posiada metodę String Execute(String A, String B) przyjmującą jako parametr dwa łańcuchy znaków oraz zwracającą jako wynik łańcuch znaków zawierający sumę liczb A i B.
- Napisz program pobierający od użytkownika dwie duże liczby jako łańcuchy znaków. Następnie na ekranie w przyjazny sposób wyświetlany jest wynik dodawania przekazanych liczb. Przykładowo:

```
432143
+
2323432
=
2755575
```

- Rozszerz klasę Addition o dodatkową metodę pozwalającą na przekazanie nazwy pliku w którym przechowywane są w kolejnych liniach duże liczby naturalnych (np. dla pliku numbers.txt). Zadaniem metody jest obliczenie sumy wszystkich liczb znajdujących się w pliku.

Podpowiedź: Przeanalizuj dodawanie pisemne liczb.

Zad. 2 Napisz program pozwalający na wczytywanie danych z pliku tekstowego oraz wyświetlający w odwrotnej kolejności na ekranie html'ową listę numerowaną. Użyj struktury danych typu stos.

Przykładowo dla danych z pliku zakupy.txt:

```
ryż x2
mąka
kisiel
foccacia
woda
```

oraz określonej nazwy klasy „shoppinglist” program wygeneruje kod:

```
<ol class="shoppinglist">
    <li>woda</li>
    <li>foccacia</li>
    <li>kisiel</li>
    <li>mąka</li>
    <li>ryż x2</li>
</ol>
```

Jak pokazano powyżej, dodaj możliwość definiowania nazwy CSS'owej klasy dla tag'a ol.

Zastosuj rozwiązanie dla plików shoppingList.txt oraz polishFootballTeam.txt.

Podpowiedź: Zdefiniuj klasę htmlList, która w polu przechowuje nazwę CSS'owej klasy tworzonej listy oraz posiada metodę String generateCode(String fileName) przyjmującą jako parametr nazwę pliku oraz zwracającą w wyniku działania szukaną html'ową listę stworzoną na podstawie danych z pliku.