
Zadanie 84.

Pobierz ciąg tekstowy z klawiatury (możliwe polskie znaki!). Następnie, używając tylko liczb dziesiętnych, spacji i/lub średnika, wymyśl sposób i utwórz funkcję zapisującą ten napis w pliku *84_kod.txt*. Utwórz funkcję odczytu treści pliku *84_kod.txt* i odwracając swoją własną metodę, zamień treść z pliku na tekst, który na początku pobrałeś z klawiatury.

Na przykład pobierasz słowo rower. Następnie zapisujesz je w jakiś sposób w pliku *84_kod.txt*, korzystając tylko z liczb dziesiętnych, spacji i/lub średnika (nie możesz wprost zapisać słowa rower). W dalszej kolejności utwórz funkcję odwracającą ten proces: przeczytaj plik *84_kod.txt* i zamień jego zawartość na słowo rower. Wyświetl

wynik na ekranie jako dowód poprawnego przeczytania i odkodowania napisu z pliku.

Zadanie 85.

Utwórz funkcję o nazwie `mCount()`, która dla podanej jako argument listy `L` z liczbami całkowitymi oraz pewnej liczby `x` szuka tej liczby `x` na liście `L` i zwraca liczbę jej wystąpień. Nie korzystaj z wbudowanej funkcji `count()`, napisz swoją wersję. Wywołaj funkcję dla dowolnej listy.

Zadanie 87.

Utwórz funkcję, która otrzymuje dwie listy z liczbami, a zwraca listę zawierającą tylko te liczby, które występują w obu listach (własny algorytm na część wspólną). Liczby występujące w zwracanej liście nie powinny się powtarzać. Możesz skorzystać z `yield`. Nie wykorzystuj zbiorów (`set`) ani funkcji `intersection()`.

Zadanie 88.

Utwórz funkcję, która przetwarza napis tak, aby zniknęły z niego wszelkie podciągi znaków powtarzających się. Zasada usuwania:

- Szukając od lewej, znajdź pierwszą parę identycznych sąsiednich znaków i usuń ją.
- Powtarzaj wcześniejszy krok aż do momentu, gdy nie zostanie znaleziona ani jedna para sąsiednich identycznych znaków.

Wykonaj zadanie dla następujących trzech napisów:

skfuu12m2111flx

npq985yy5vnpr8ettemzp9yamppp98tywnmtvb9q8y4nqp948tydlfaaafd

1abbbbaaccdccdeeddcaeffabcdefa121122

Utwórz drugą funkcję z tą różnicą, że dwa identyczne sąsiednie znaki będzie zamieniać na jeden, np. `xx` zamieni na `x`. Sprawdź wynik obu funkcji dla tych samych napisów.

Zadanie 89.

Utwórz funkcję, która dla podanej liczby całkowitej dodatniej zwraca True, jeśli liczba jest pierwsza, lub False w przeciwnym razie. Przyjmij, że 1 nie jest liczbą pierwszą. Wygeneruj 1000 liczb losowych z przedziału <2;1000>. Policz ilość liczb pierwszych, które udało się wylosować. Wykorzystaj utworzoną funkcję.

Zadanie 90.

Utwórz funkcję, która przekazane dwie listy z liczbami (float) łączy w jedną listę i zwraca ją. Liczby na liście wynikowej powinny być ułożone naprzemiennie: raz liczba z pierwszej listy, raz z drugiej. Liczby do nowej listy pobieraj z istniejących list — zaczynaj od początku, kieruj się ku końcowi (od lewej do prawej). Gdy liczby na jakiejś liście się wyczerpią, niewykorzystane liczby z drugiej listy mają zostać dołączone na koniec listy zwracanej. Uruchom swoją funkcję dla dwóch list:

L1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

L2 = [11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24]