

Wstęp do programowania 2017-2018

Pracownia 11

Uwaga: Niezależnie od sytuacji w danej grupie zadania z tej listy można oddawać za 100% co najmniej do drugich zajęć w 2017. Na tej liście znowu będą wprawki o tematyce wybranej przez prowadzącego ćwiczenia. Zadania z gwiazdką z tej listy będą ważne do końca semestru. Wszystkie zadania która pojawią się **po** tej liście mają termin ważności do końca semestru.

Premia za tę listę wynosi 0.5, przyznawana jest osobom, które zdobyły co najmniej 2p za zadania z tej listy. Maksimum dla tej listy wynosi 4p.

Zadanie 1.(1pkt) Szyfrowanie metodą Cezara polega na tym, że każdą literę danego słowa zamienia się na literę przesuniętą o k pozycji (zgodnie z porządkiem alfabetycznym, w którym po ostatniej literze (z) następuje litera pierwsza (a)). W szyfrze Cezara kluczem umożliwiającym szyfrowanie (i odszyfrowanie) jest liczba k .

Napisz funkcję `ceasar(s,k)`, która dla danego słowa s i klucza k znajduje wartość szyfrogramu. Pamiętaj, by używać polskiego alfabetu (aąbcćdeęfghijklmńńóóprśśtuwyzzzz). Zastanów się, jak można byłoby tu wykorzystać słowniki i funkcję `zip` do utworzenia zwięzlejszego i eleganckiego kodu.

Zadanie 2.(1pkt) Parę słów nazwiemy *parą cesarską* (a występujące w niej słowa *cesarskimi*), jeżeli są one wzajemnie swoimi szyfrogramami w szyfrze Cezara (tzn. każde z nich otrzymujemy z drugiego za pomocą odpowiedniego przesunięcia wszystkich liter; oczywiście przesunięcie powinno być nietrywialne, czyli nie może być identycznością). Napisz program, który znajduje najdłuższe polskie słowo cesarskie (jeżeli więcej niż jedno osiąga maksymalną długość powinieneś wypisać je wszystkie).¹

Zadanie 3.(1+Xpkt) Zmodyfikuj program `komiwojazer.py` (na SKOS) tak, żeby był w stanie wczytać punkty z pliku i znaleźć możliwie krótką drogę dla Mikołaja. Czyli interesuje nas permutacja punktów, a jej wartością jest suma odległości między kolejnymi punktami. Żeby zadanie było zaliczone, droga policzona dla zestawu punktów na SKOS powinna mieć długość mniejszą niż 5300 jednostek. Da się to zrealizować prostym zachłannym algorytmem (spróbuj go wymyśleć samodzielnie). Algorytm dający ten wynik jest następujący: `Cemrtyąqnl eózar chaxgl fgnegbjr. Qyn xnzqrtb m avpu jlmanpmnzl "mnpulnaaa"qebtę, j xgóerw m xnzqrtb chaxgh qvmvrzl qb anwoy- vzfmrtrb avrbqjvrqmbartb qb grw cbel.`

Premię można otrzymać za znalezienie drogi krótszej niż 5257. Dokładne zasady naliczania tej premii będą podane później, na pewno osoba, która znajdzie najkrótszą drogę, krótszą niż 5257 może liczyć na co najmniej dodatkowe 2 punkty.

Zadanie 4.(1pkt) Zdefiniujmy następujące przekształcenie na słowach (nazwiemy je *permutacyjną postacią normalną*): zamieniamy litery na liczby, w ten sposób, że:

1. Tym samym literom przypisane są równe liczby, różnym literom – różne liczby.
2. Liczby przypisywane są po kolei, licząc od lewej strony.

Otrzymane liczby sklejamy w jeden napis, wstawiając na przykład znak "-" jako separator. Przykładowe pary słowo i wartość przekształcenia: tak: 1-2-3, nie: 1-2-3, tata: 1-2-1-2, indianin: 1-2-3-1-4-2-1-2. Napisz funkcję, która zwraca w wyniku wartość opisanego przekształcenia.

Zadanie 5.(1pkt) Szyfr przestawieniowy to taki szyfr, w którym każdej literce z polskiego alfabetu przypisana jest inna literka (konsekwentnie, w ramach całego komunikatu). W tym i kolejnym zadaniu, będziemy łamać takie szyfry (czyli pisać programy, które znajdują komunikat, w sytuacji, gdy mamy znany jedynie szyfrogram). Będziemy zakładać, że słowa w szyfrogramie oddzielone są spacjami i (dla zwiększenia czytelności komunikatu), między nimi czasami znajdują się znaki interpunkcyjne (niezaszyfrowane, otoczone spacjami). Zakładamy również, że wszystkie słowa w

¹W pewnej poprzedniej edycji tego przedmiotu było zadanie, w którym należało odszyfrować listy do Świętego Mikołaja zaszyfrowane szyfrem Cezara z nieznanym k (cały list z jednym k). Okazuje się, że istnieją szyfrogramy, składające się z kilku wyrazów, które da się zinterpretować jako życzenia, które można odczytać na więcej niż 1 sposób. Czy umiesz jakieś wskazać?

komunikacie występują w słowniku (z polskimi słowami z jednej z poprzednich list) i że nie mamy żadnych dodatkowych informacji o języku (np. o częstościach liter, czy wyrazów).

Napisz program, który umie rozszyfrować dwa pierwsze szyfrogramy ze SKOS-u. Uwaga: w obu tych szyfrogramach wszystkie słowa mają unikalną permutacyjną postać normalną (to znaczy, że znajomość tejże postaci pozwala jednoznacznie wybrać słowo). Uwaga2: każdy szyfrogram jest w osobnym wierszu, każdy był też szyfrowany osobną permutacją.

Zadanie 6.(0.5, ★pkt) Zmodyfikuj program, by poradził sobie z jeszcze jakimś przykładem, w którym nie wszystkie słowa mają unikalną permutacyjną postać normalną (na przykład trzeci szyfrogram ma tę własność, będąc zarazem bardzo łatwym do odszyfrowania)

Zadanie 7.(1, ★pkt) Zmodyfikuj program, by poradził sobie z wszystkimi przykładami ze SKOS-u. Uwaga: to już trochę trudniejsze zadanie. W rozwiązaniu można wykorzystać fakt, że nawet nie-unikalna normalna postać permutacyjna daje pewne wskazówki o możliwych przyporządkowaniach liter. Szyfrogram 'óśóś' mówi na przykład, że literze 'ó' nie można przypisać litery 'ą' (bo żadne słowo nie zaczyna się na 'ą'). Można też z niego wydedukować inne rzeczy, patrząc na wszystkie słowa o postaci permutacyjnej 1-2-1-2.

Szyfrogramy ze SKOS-u da się odszyfrować w czasie kilku sekund każdy (a niektóre w ułamki sekundy).