

Algorytmy i Struktury Danych

Kolokwium 1 (23. IV 2020)

Format rozwiązań

Rozwiązanie każdego zadania powinno być wysłane jako pojedynczy plik tekstowy (rozszerzenie txt). W pierwszej linii należy podać swoje imię i nazwisko. Rozwiązania w innych formatach (np. PDF, DOC, PNG, JPG) z definicji nie będą sprawdzane i otrzymają ocenę 0 pkt. nawet jeśli będą poprawne. W szczególności niedopuszczalne jest rozwiązanie zadania na kartce i wysłanie jej zdjęcia.

Python

Kod w Pythonie powinien korzystać tylko z elementarnych fragmentów języka. Niedopuszczalne jest korzystanie z wbudowanych funkcji sortujących, słowników, zbiorów, jakichkolwiek operacji import. Wszystkie użyte struktury danych należy zaimplementować samodzielnie (poza listami Pythonowymi traktowanymi jako tablice; np. $A = [0] * n$ jest dopuszczalne).

Zadania

[2pkt.] Zadanie 1. Cyfra jednokrotna to taka, która występuje w danej liczbie dokładnie jeden raz. Cyfra wielokrotna to taka, która w liczbie występuje więcej niż jeden raz. Mówimy, że liczba naturalna A jest ładniejsza od liczby naturalnej B jeżeli w liczbie A występuje więcej cyfr jednokrotnych niż w B , a jeżeli cyfr jednokrotnych jest tyle samo to ładniejsza jest ta liczba, która posiada mniej cyfr wielokrotnych. Na przykład: liczba 123 jest ładniejsza od 455, liczba 1266 jest ładniejsza od 114577, a liczby 2344 i 67333 są jednakowo ładne.

Dana jest tablica T zawierająca liczby naturalne. Proszę zaimplementować funkcję:

`pretty_sort(T)`

która sortuje elementy tablicy T od najładniejszych do najmniej ładnych. Użyty algorytm powinien być możliwie jak najszybszy. Proszę w rozwiązaniu umieścić 1-2 zdaniowy opis algorytmu oraz proszę oszacować jego złożoność czasową.

[2pkt.] Zadanie 2. Mamy n żołnierzy różnego wzrostu i nieuporządkowaną tablicę, w której podano wzrosty żołnierzy. Żołnierze zostaną ustawieni na placu w szeregu malejąco względem wzrostu. Proszę zaimplementować funkcję:

`section(T,p,q)`

która zwróci tablicę ze wzrostami żołnierzy na pozycjach od p do q włącznie. Użyty algorytm powinien być możliwie jak najszybszy. Proszę w rozwiązaniu umieścić 1-2 zdaniowy opis algorytmu oraz proszę oszacować jego złożoność czasową.

[2pkt.] Zadanie 3. Proszę zaproponować algorytm, który dla tablicy liczb całkowitych rozstrzyga czy każda liczba z tablicy jest sumą dwóch innych liczb z tablicy. Zaproponowany algorytm powinien być możliwie jak najszybszy. Proszę oszacować jego złożoność obliczeniową.