

## Kolokwium 2

### Zadanie 1 (2 pkt)

Napisz skrypt, który odczyta tekst z pliku i sprawdzi, czy dana linia tekstu jest palindromem. Jeżeli jest, zapisze ją do pliku „palindromy.txt”, a jeżeli nie, zapisze ją do pliku „nie\_palindromy.txt”. Nazwa pliku z tekstem do zbadania powinna być podana jako pierwszy i jedyny argument wiersza poleceń. Gdy plik nie istnieje skrypt powinien wyświetlić stosowny komunikat i zakończyć działanie. Program powinien wykorzystywać konstrukcję „try-except” oraz „with-open”.

### Zadanie 2 (4 pkt)

Napisz trzy funkcje. Pierwsza niech dla zadanej liczby naturalnej zwraca listę jej dzielników. Druga niech sprawdza, czy podana liczba jest liczbą pierwszą. Trzecia niech sprawdza, czy podana liczba jest liczbą doskonałą. [https://en.wikipedia.org/wiki/Perfect\\_number](https://en.wikipedia.org/wiki/Perfect_number) . Niech program prosi użytkownika o podanie liczby, sprawdza czy użytkownik faktycznie podał liczbę naturalną (użyj konstrukcji „try-except”) i wypisze w konsoli jej dzielniki oraz stosowny komunikat, jeżeli liczba okazała się liczbą pierwszą i/lub liczbą doskonałą.

### Zadanie 3 (4 pkt)

Stwórz moduł sortowanie.py zawierający dwie funkcje, które przyjmują nieposortowaną listę jako argument wejściowy i zwracają listę posortowaną. Każda z funkcji ma korzystać z innej metody sortowania. Możliwe metody można znaleźć tutaj: <https://cpp0x.pl/kursy/Algorytmy/Sortowanie-danych/443> . Nie korzystaj z wbudowanej funkcji sort() oraz sorted()! Informacja o wybranej metodzie ma się znaleźć w komentarzu wewnątrz funkcji (dla chętnych – możecie użyć docstringa). W osobnym pliku napisz skrypt, który tworzy listę zawierającą 1000 losowych liczb i sortuje tą listę używając funkcji zdefiniowanych w module sortowanie.py . Skrypt ma wypisać w konsoli czas sortowania listy jedną i drugą funkcją.

### Zadanie 4 (8 pkt)

Napisz klasę „Obywatel” której konstruktor przyjmuje parametry: „imie”, „nazwisko” i „data urodzenia”. Na podstawie daty urodzenia ustalany jest wiek (skrypt ma sam pobrać aktualną datę i porównać z datą urodzenia!). W klasie należy zdefiniować metodę „\_\_str\_\_” która zwraca stringa o treści: „Obywatel/ka (imie) (nazwisko), (wiek) lat, urodzony (data urodzenia)”. Należy zademonstrować działanie metody \_\_str\_\_ poprzez użycie funkcji „print(osob)” gdzie „osob” to instancja klasy „Obywatel”.

**Punktacja ocen:**

16 - 18 : 5.0

14 - 15 : 4.5

11 - 13 : 4.0

09 - 10 : 3.5

06 - 08 : 3.0