

Zadania do zrobienia w Pythonie :)

Alicja Hołowiecka

1 Zadania z programowania

Te zadania pochodzą ze zbioru zadań *Python, C++, JavaScript. Zadania z programowania* Marka Lulińskiego i Gniewomira Sarbickiego.

1. Liczba jest parzysta, jeżeli dzieli się przez 2. Innym sposobem rozpoznania jest sprawdzenie jej ostatniej cyfry - musi to być 0, 2, 4, 6 albo 8.
Jeżeli podana liczba jest większa od 1, to możemy odejmować od niej liczbę 2 tak długo, aż uzyskamy 0 albo 1 i w ten sposób rozstrzygniemy o parzystości.
Sprawdź na kilka sposobów, czy podana liczba naturalna jest parzysta.
2. Liczba jest pierwsza, jeśli posiada dokładnie dwa dzielniki.
Sprawdź, czy podana liczba naturalna jest liczbą pierwszą.
3. Liczba jest super-pierwsza, jeśli jest liczbą pierwszą oraz suma jej cyfr też jest liczbą pierwszą.
Taką liczbą jest np. 101.
Sprawdź, czy podana liczba naturalna jest liczbą super-pierwszą.
4. Liczba jest B-super-pierwsza, jeśli jest liczbą super-pierwszą oraz suma dziesiętna jej cyfr w rozwinięciu dwójkowym jest też liczbą pierwszą.
Taką liczbą jest np. 1291.
Sprawdź czy podana liczba naturalna jest B-super-pierwsza.
5. Liczba doskonała - liczba naturalna, która jest sumą wszystkich swoich dzielników właściwych (tzn. mniejszych od niej).
Najmniejszą liczbą doskonałą jest 6, ponieważ $6 = 3 + 2 + 1$. Następna to 28 ($28 = 14 + 7 + 4 + 2 + 1$).
Sprawdź, czy podana liczba naturalna jest liczbą doskonałą.
6. Liczba podzielna - liczba naturalna, która jest większa od 0 i dzieli się przez sumę swoich cyfr.
Taką liczbą jest np. 21.
Sprawdź, czy podana liczba naturalna jest liczbą podzielną.

7. Liczby Mersenne'a to liczby postaci $2^p - 1$, gdzie p jest liczbą pierwszą.
Sprawdź, czy podana liczba naturalna jest liczbą Mersenne'a.
8. Liczby bliźniacze - takie dwie liczby pierwsze, których różnica wynosi 2.
Przykłady liczb bliźniaczych: 3 i 5, 5 i 7, 11 i 13.
Sprawdź, czy podane dwie liczby naturalne są liczbami bliźniaczymi.
9. Liczby zaprzyjaźnione to para różnych liczb naturalnych, takich, że suma dzielników właściwych każdej z tych liczb równa się drugiej (dzielniki właściwe są to wszystkie dzielniki danej liczby oprócz niej samej).
Przykłady liczb zaprzyjaźnionych: 220 i 284, 1184 i 1210, 2620 i 2924, 5020 i 5564.
Sprawdź, czy podane dwie liczby naturalne są liczbami zaprzyjaźnionymi.
10. Algorytm Euklidesa - algorytm wyznacza największy wspólny dzielnik dwóch liczb całkowitych dodatnich.
Algorytm opiera się na dwóch faktach:
 - największy wspólny dzielnik pewnej liczby i zera to ta liczba
 - największy wspólny dzielnik liczb a i b jest równy największemu wspólnemu dzielnikowi liczb $(b \bmod a)$ i a .

Pierwsze wzmianki na temat tego algorytmu pojawiły się w dziele Euklidesa zatytułowanym *Elementy* około 300 lat przed naszą erą. Istnieją różne wersje tego algorytmu.

Teraz czas na Ciebie - znajdź największy wspólny dzielnik dwóch podanych liczb naturalnych!

2 Zadania na danych pokemon_data

1. Do wykonania jako notatnik .ipynb (Jupyter Notebook).
Wczytaj dane `pokemon_data.csv`.
Napisz raport, w którym przedstawisz:
 - oczyszczanie danych
 - podstawowe statystyki opisowe (średnia, mediana, odchylenie standardowe, kwartyle)
 - sortowanie danych: alfabetycznie według 'Type 1' i jednocześnie rosnąco według 'HP'
 - filtrowanie danych: wybierz pokemony które mają 'Type 1' Grass, 'Type 2' Poisson oraz 'HP' większe niż 70. Zapisz te dane do plików .csv, .xlsx oraz .txt.
 - usunięcie z tabeli kolumny z numerem Pokemona

- zliczenie ile jest pokemonów danego Typu 1 (wyświetlić ma się nazwa Typu oraz liczebność), oraz ile jest Pokemonów danego Typu 2 wewnątrz każdego Typu 1 (np. ile jest Pokemonów Typu 2 'Poison' wśród Pokemonów Typu 1 'Bug' itd.)
- zmianę w kolumnie 'Legendary' na 'True' dla wszystkich Pokemonów, które są typu ognistego oraz mają 'HP' większe niż 100
- zapisanie do osobnego pliku tabeli z mega ewolucjami (Pokemony mające w nazwie 'Mega'), która będzie zawierać kolumny: Name, Type 1, Type 2. Sprawdź, w jakim typie jest najczęściej mega ewolucji

Postaraj się jak najdokładniej opisać wykonanie każdego z poleceń. Raport zapisz jako notatnik jupyter, a następnie spróbuj przekonwertować go do plików .html, .py, .tex oraz .pdf.

2. Drużyna Pokemonów składa się z 6 Pokemonów, w tym maksymalnie jednej mega ewolucji (zawiera 'Mega' w nazwie). Nie mogą to być pokemony **Legendary**. Liderzy Pokemon mają całą drużynę złożoną z Pokemonów jednego typu (**Type 1**), dla pozostałych trenerów jest to obojętne.

Napisz program, który wczyta dane `pokemon_data.csv`. Do tabeli dodaj kolumnę **Total**, która będzie sumą wszystkich statystyk danego Pokemona (kolumny **HP**, **Attack** itd.). Program ma wybrać jak najlepszą drużynę Pokemonów, na podstawie kolumny **Total**, w zależności od tego, czy trener jest liderem czy nie, i czy chce mieć w swojej drużynie mega ewolucję.

Program powinien wyświetlić w konsoli wybraną drużynę : sześć wierszy, w każdym z nich imię Pokemona, jego oba typy oraz **Total**.

Do wykonania jako skrypt .py (np. w IDLE, PyCharm, Spyder itp.) lub jako notatnik .ipynb (w Jupyter Notebook).

