

Język Python

[Strona główna](#) / [Moje kursy](#) / [JP](#) / [Sekcja ogólna](#) / [LAB 10 - sr. 15:30](#)

LAB 10 - sr. 15:30

1. Proszę utworzyć funkcję sprawdzającą poprawność numeru PESEL (3.5p)

Parametrami wejściowymi do funkcji są: Pesel, data urodzenia (datetime.date) oraz płeć.

Przykłady:

- 02070803628, 8 lipca 1902, kobieta
- 02270803624, 8 lipca 2002, kobieta
- 02270812350, 8 lipca 2002, mężczyzna

PESEL

- cyfry 1-2 to ostatnie dwie cyfry roku urodzenia
- cyfry 3-4 to dwie cyfry miesiąca urodzenia
- cyfry 5-6 to dwie cyfry dnia urodzenia
- cyfry 7-10 liczba porządkowa z oznaczeniem płci (liczba parzysta - kobieta, liczba nieparzysta - mężczyzna)
- cyfra 11 suma kontrolna

Do numeru miesiąca dodawane są następujące wartości w zależności od roku:

- dla lat 1800 - 1899 - 80
- dla lat 1900 - 1999 - 0
- dla lat 2000 - 2099 - 20
- dla lat 2100 - 2199 - 40
- dla lat 2200 - 2299 - 60

Suma kontrolna: każdą pozycję numeru ewidencyjnego mnoży się przez odpowiednią wagę, są to kolejno: 1 3 7 9 1 3 7 9 1 3 i sumuje.

Wynik dzieli się modulo 10 i otrzymaną wartość należy odjąć od 10 i znów podzielić przez modulo 10.

Otrzymana wartość powinna być zgodna z ostatnią cyfrą numeru PESEL.

2. Proszę napisać funkcję zwracającą średni wiek osób, który daty urodzenia zapisane są w plik *daty.in*. (3.5p)

Funkcja powinna móc działać w trybie 'restrykcyjnym' - po napotkaniu niepoprawnej daty/wpisu zgłoszenie wyjątku i zakończenie działania, w trybie 'liberalnym' - niepoprawne wpisy są ignorowane.

Linia w pliku jest poprawna, jeśli zawiera dzień, miesiąc i rok, które tworzą poprawną datę - zgodność liczby dni w miesiącu, w tym odpowiednia długość lutego w zależności od tego czy rok jest przestępny czy nie.

Rok przestępny: podzielny przez 4 i niepodzielny przez 100 lub podzielny przez 400.

3. Proszę napisać funkcję sprawdzającą czy elementy listy tworzą trójkę ($a^2+b^2=c^2$)/czwórkę ($a^2+b^2+c^2=d^2$) pitagorską (funkcja ma działać dla dowolnej długości "podciągu"!)). Proszę zgłosić wyjątek w przypadku niepoprawnej długości listy oraz w przypadku, gdy lista nie zawiera żadnych trójek/czwórek pitagorejskich. Dla każdej trójki/czwórki proszę sprawdzić ile jest w niej wartości parzystych i nieparzystych (3p).

Listy testowe:

$l =$

(1,2,2,3,2,3,6,7,1,4,8,9,4,4,7,9,2,6,9,13,6,6,7,11,3,4,12,13,2,5,14,15,2,10,11,15,1,12,12,17,8,9,12,17,1,6,18,19,6,6,17,19,6,10,15,21,4,5,20,21,4,8,19,21

$l =$

(1,2,2,3,2,3,6,7,1,4,8,9,4,4,7,9,2,6,9,13,6,6,7,11,3,4,12,13,2,5,14,15,2,10,11,15,1,12,12,17,8,9,12,17,1,6,18,19,6,6,17,19,6,10,15,21,4,5,20,21,4,8,19,21

$l = (3,4,5,5,12,13,7,24,25,9,40,41,6,8,10,60,80,100,18,24,30,15,8,17)$



$l = (1,2,3,4,5,6,7,8,9)$

4 maja 2021, 16:19

Status przesłanego zadania

Status przesłanego zadania	Przesłane do oceny	
Stan oceniania	Ocenione	
Termin oddania	środa, 5 maja 2021, 17:45	
Pozostały czas	Zadanie zostało złożone 1 min 22 sek. przed terminem	
Ostatnio modyfikowane	środa, 5 maja 2021, 17:43	
Przesyłane pliki	 TomaszSzkarakdekPython.zip 5 maja 2021, 17:43	
Komentarz do przesłanego zadania	 Komentarze (0)	

Informacja zwrotna

Ocena	9,00 / 10,00
Ocenione dnia	czwartek, 6 maja 2021, 21:33
Ocenione przez	 Małgorzata Krawczyk
Komentarz zwrotny	<div> zad1 - wywołanie f-cji mogących zgłaszać wyjątki dobrze umieszczać w blokach try zad3 - miało działać dla dowolnej długości podciągu, nie ...</div>

◀ ŚRODA 15:30

Przejdź do...

Quiz 10 - śr. 15:30 ▶



Platforma e-Learningowa obsługiwana jest przez:
Centrum e-Learningu AGH oraz Centrum Rozwiązań Informatycznych AGH

[Pobierz aplikację mobilną](#)