

Python (11) - czas i data

(1) Moduł *time* zawiera funkcje związane z czasem i datami. Przetestuj klikukrotnie funkcję *time()* i przeanalizuj wyniki, np.:

```
import time
i=time.time()
print(i)
```

Stosując *time()* napisz funkcję *czekaj(s)* która powoduje, że program 'nie robi' przez *s* sekund. Przetestuj.

Sprawdź, że funkcja *sleep(s)* działa tak samo jak *czekaj*.

(2) Moduł *datetime* jest również przydatny do czasu i dat. Przetestuj:

```
import datetime
dzis=datetime.date.today()
print(dzis.timetuple())
d=dzis.timetuple()
print(d.tm_year)
```

Możesz zmieniać datę (*dzis*) za pomocą, np.:

```
innadata=datetime.date(2015,12,30)
```

Przeanalizuj pola *innadata.timetuple()*. Znajdź pole dotyczące dnia tygodnia i napisz program w którym użytkownik wprowadza datę a program wyświetla jaki to był dzień tygodnia (po polsku).

(3) Stwórz datę *wczoraj* za pomocą *date*. Przetestuj działanie funkcji *mktime* na *dzis.timetuple()* z poprzedniego zadania oraz na *wczoraj.timetuple()*:

```
t=time.mktime(dzis.timetuple())
tt=time.mktime(wczoraj.timetuple())
```

Badając *t – tt* domyśl się co zwraca *mktime*.

(4) Przetestuj działanie funkcji *ctime*, np.:

```
t2=time.ctime(t)
```

Napisz program który wyświetla daty i czasy *k* sekund temu gdzie *k* jest 10 tysięcy, 100 tysięcy, milion, itd. Sprawdź do jakich wartości program działa. Przeanalizuj wartość zwracaną przez *time.time()* z pierwszego zadania. Od jakiego czasu jest mierzona zwracana ilość sekund?

(5) Napisz program w którym wyświetlona jest losowa liczba *k* od 1 do 10. Następnie użytkownik wciska ENTER, czeka około *k* sekund i wciska znowu ENTER. Program wylicza i wyświetla o ile się użytkownik pomylił,

np:

2.3 sekundy za wcześnie

0.5 sekundy za późno

(6) (dodatkowe) Do/od daty można dodawać/odejmować różnice czasowe, np.:

2

```
dl=datetime.timedelta(days=1)
print(dzis+dl)
print(dzis-dl)
```

Napisz program który podaje datę najbliższej ostatniej niedzieli w miesiącu.
Dodatkowo podane jest za ile dni będzie ta niedziela.