

1. domácí úkol

- zadaný: 2. 10. 2023
- deadline: 9. 10. 2023, **23:55**

Pokyny

- Odevzdejte jediný soubor pojmenovaný `prijmeni_du1.py` (“prijmeni” nahradte svým příjmením) s definicemi požadovaných funkcí. Odevzdávárnu najdete v ISu: Student / FI:IB113 / Odevzdávárny / Skupina 10 / [DU1](#)
- Šablona řešení je dostupná [ve studijních materiálech](#).
- Jména odevzdávaných funkcí neměňte; ať zůstanou pojmenované stejně, jako jsou pojmenované v zadání.
- Případné dotazy pokládejte v diskuzním fóru této skupiny [ve vláknu k tomuto úkolu](#).
- Příklad vypracujte samostatně. Nepovolená spolupráce bude potrestána dle pravidel předmětu.

Příklady

Posloupnost

Napište funkci `sum_sequence(n)`, která vypíše prvních n členů posloupnost, jejíž k -tý prvek je součet prvních k přirozených čísel. Parametr n bude vždy kladné celé číslo. Členy posloupnosti nechtě jsou odděleny právě jednou mezerou.

```
>>> sum_sequence(6)
1 3 6 10 15 21
```

Srovnávání čísel

Napište funkci `compare(n1, n2)`, která porovná dvě předaná celá čísla. Pokud jsou si čísla rovna, funkce vypíše `same`, pokud jsou různá, funkce vypíše větší z nich a informaci o tom, o kolik je větší. Znění hlášek zkuste dodržet (až na konkrétní čísla) doslova stejně s příklady níže.

```
>>> compare(6, 10)
10 is greater by 4
>>> compare(6, 6)
same
```

Textová grafika: písmeno D

Napište funkci `letterD(height)`, která pomocí textové grafiky vypíše písmeno D zadané výšky. Maximální šířka budiž 5 znaků.

Vykreslené písmeno D začíná dvěma mřížkami `##` a každý další řádek ve vrchní polovině písmene obsahuje mezi mřížkami o tečku víc než předchozí, až do maximální šířky. Po dosažení maximální šířky už se počet teček jenom opakuje. Spodní polovina písmene D je symetrická. Pro vyjasnění se podívejte na příklady níže.

Zkuste se zamyslet, jak byste mohli použít funkci `min` pro zjištění, kolik teček má být před danou mřížkou.

```
>>> letterD(4)
```

```
##  
#.#  
#.#  
##
```

```
>>> letterD(7)
```

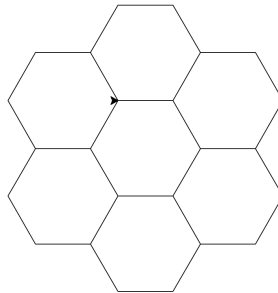
```
##  
#.#  
#..#  
#...#  
#..#  
#.#  
##
```

```
>>> letterD(10)
```

```
##  
#.#  
#..#  
#...#  
#...#  
#...#  
#...#  
#..#  
#.#  
##
```

Želví grafika: včelí plástev

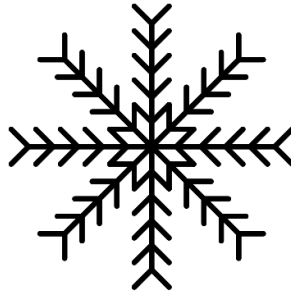
Napište funkci `honeycomb(length)`, která vykreslí pomocí želví grafiky včelí plástev podle následující ukázky. Parametr `length` je kladné celé číslo vyjadřující délku hrany šestiúhelníku, který plástev tvoří.



Ukázka (finální pozici želvy není nutné dodržet)

Želví grafika: vločky

Napište funkci `snowflake(n, length)`, která vykreslí vločku podle následujícího vzoru. Parametr `n` vyjadřuje počet drobnějších paprsků na každém z osmi hlavních paprsků. Drobnější paprsky by měly být rovnoměrně rozloženy podél hlavního paprsku, jehož délka je `length`. Délku drobnějších paprsků dopočítejte, tak, aby se akorát dotýkaly (jako v ukázce).



Ukázka

Vytvořte vlastní funkci `my_snowflake`, která bude vykreslovat sněhové vločky podle vašeho uvážení a preferencí. Funkce může mít libovolný počet parametrů (i nulový). Pro inspiraci koukněte třeba [sem](#) nebo [sem](#).