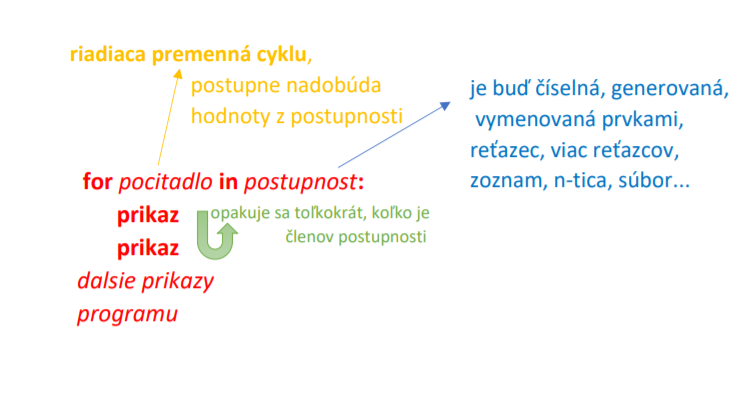
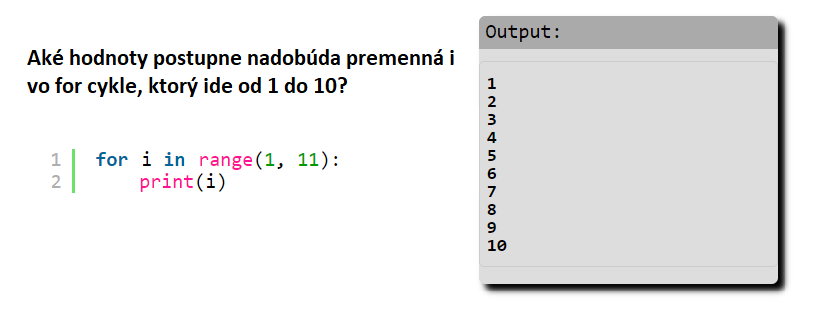
**For cyklus:**



* **i** je riadiacia premenná cyklu – počítadlo – postupne sa počas cyklu mení, zmení sa toľkokrát, koľkokrát má ísť cyklus, podľa toho, akú máme postupnosť; označenie i sa používa všeobecne, ale môže mať aj iné označenie
* pozor! range ide vždy o jedno menej. príklad: range(0,9) , prípadne toto isté viem zapísať aj takto: range(9)
  + cyklus zbehne 9-krát, ide od 0 do 8
  + teda i bude mať hodnoty 0, 1, 2, 3, 4, 5, 6, 7, 8



**Ako zapisujeme FOR cyklus v pythone? Je na to viac rôznych spôsobov:**

***Cyklus s použitím range, ktorý beží určitý počet krát:***

range(po) #napr. range(5) -> zbehne 5x, i bude mať hodnoty 0,1,2,3,4 (teda zbehne od 0 do po-1, teda do 4)

range(od, po) #napr. range(1,5) -> zbehne 4x, i bude mať hodnoty 1, 2, 3, 4

range(od, po, krok) #napr. range(1,8,2) -> zbehne 4x, i bude mať hodnoty 1, 3, 5, 7 -> preskakuje, krok hovorí ktoré číslo má preskočiť, tu každé druhé, vypíše len nepárne čísla

**#pozor! po v range, ide vždy o jedno menej!!!!**

*Zápis:*

**for i in range(10)**: #čísla od 0 do 9

print(i)

**for i in range(5,10)**: #čísla od 5 do 9

print(i, end=' ')

for i in range(2,100,2): #čísla od 2 do 99 ale iba párne - tretie číslo hovorí o preskovaní v cykle

print(i, end=' ')

-------------------------------------------------------

**for i in premenna:** #v premennaej musí myť string, pole alebo ntica napr. premenna = "Ahoj"

print(i)

for i in 1,2,3:

print(i)

for i in 2,3,5,11:

print(i\*\*2) #počítanie druhej mocniny číslu

for i in "AHOJ":

print(i, end=' ')

for i in "a","b","c","d":

print(i)

**break** #príkaz na vyskočenie z cykla, ukončenie cykla

**pass** #prázdniny príkaz, nerobí nič

**continue** #na tomto mieste sa ukončí vykonávanie danej iterácie cyklu a pokračuje sa ďalšou hodnotou riadiacej premennej

- veľmi dôležité: príkazy, ktoré sa majú vykonať v rámci for cyklu, musia byť ďalej od okraja

- oddelenie od okraja sa urobí automaticky, ak je správne zapísaná hlavička for cyklu

- prípadne vieme urobiť oddelenie - na stlačenie TAB (tabulátora)

- alebo na 4 údery medzerníkom

Úlohy:

1. Vypočítajte n! (súčin čísel 1 \* 2 \* 3 \* ... \* n.)

napr.

zadaj cislo: 6

6! = 720

2. Naprogramuj funkciu, ktorá pre dané a, b – kladné čísla, vykreslí obdĺžnik zo znakov \*. Pričom a – šírka obdĺžnika ( počet \* na šírku), b – výška ( počet \* na výšku) Napr. pre a=8 a b=4 sa vykreslí:

\* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \*

3. Vykresli pravouhlý trojuholník z daného znaku a s danou dĺžkou základne (počet \* na základni) n. Pre n=5 sa vykreslí nasledujúci trojuholník:

\*

\* \*

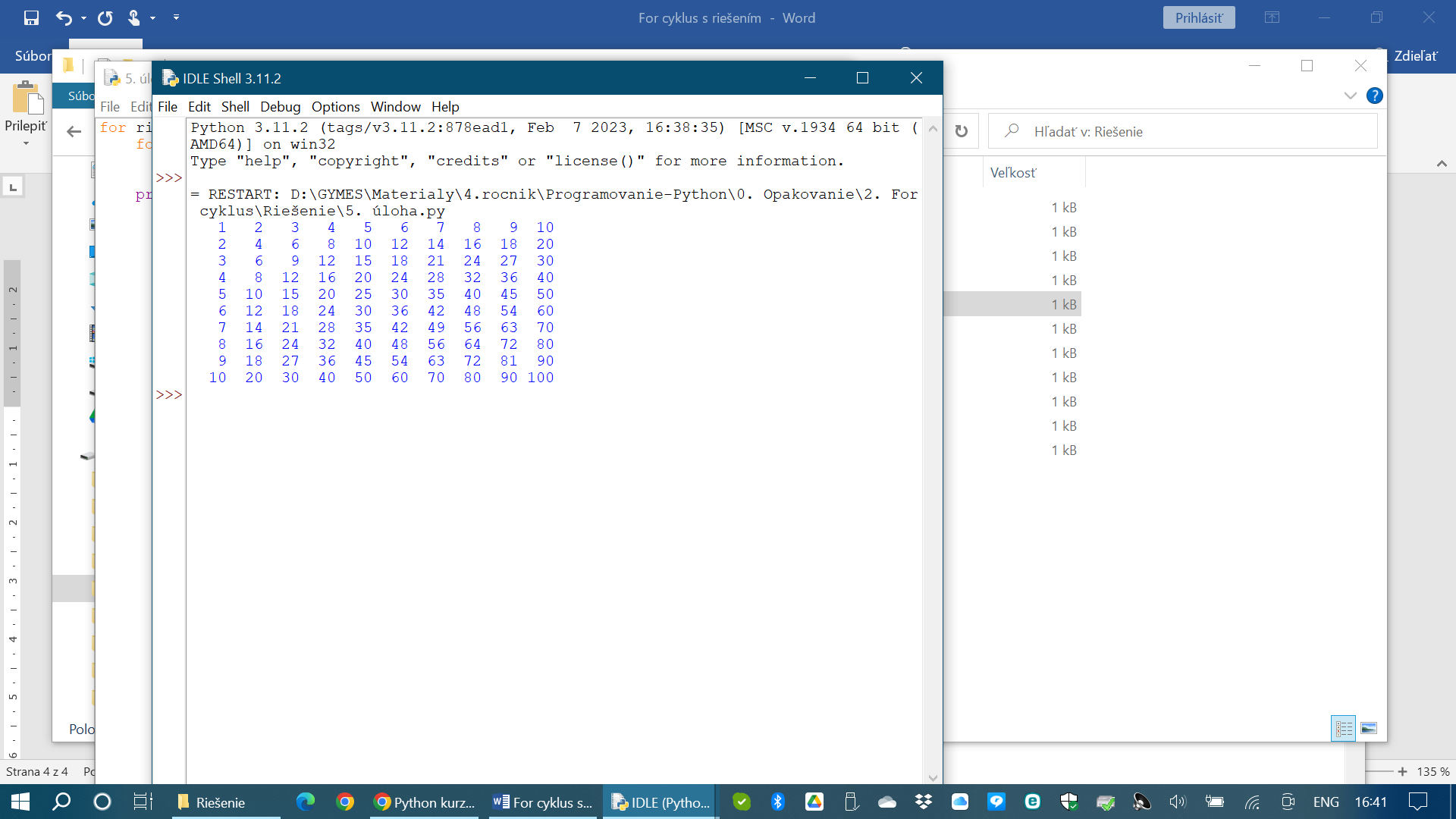
\* \* \*

\* \* \* \*

\* \* \* \* \*

4. Napíš program, ktorý každý znak daného reťazca vypíše n-krát . Napr. pre n=4 a pre ret=’ahoj’ sa vypíše:

a a a a  
 h h h h  
 o o o o  
 j j j j

5. Vypíšte na obrazovku tabuľku malej násobilky. V záhlaví riadkov aj stĺpcov budú čísla 1 až 10. V bunkách tabuľky bude súčin týchto čísel.

Tip na riešenie:

❖ Aby bol výstup pekne sformátovaný použijeme vo funkcii print metódu format, ktorá

vytvára formátovaný reťazec.

o "{:4}".format(sucin). Formátovací reťazec {:4} znamená, že na

miesto zátvoriek sa doplní parameter metódy format a doplní sa medzerami

na 4 znaky.

❖ V príkaze print priradenie prázdneho reťazca parametru end spôsobí, že po výpise

sa neprejde na nový riadok. (Parameteru end môžeme priradiť ľubovoľný reťazec,

ktorým sa výpis ukončí).

6. Vytvorte program, ktorý vypíše všetky delitele zadaného prirodzeného čísla.

Tip na riešenie: Deliteľ čísla x je číslo d, ktoré delí číslo x bezo zvyšku (zvyšok je nula) x % d == 0

Domáca úloha:

7. Vytvorte program, ktorý zistí a vypíše, či je zadané celé číslo prvočíslo.

8. Používateľ zadá slovo, všetky samohlásky v slove nahraďte hviezdičkou a nové slovo s hviezdičkami vypíšte.

9. Naprogramuj úlohu 3 tak, aby sa trojuholník vykresľoval „dolu hlavou“. T.j pre n=5

\* \* \* \* \*

\* \* \* \*

\* \* \*

\* \*

\*

10. Vytvorte program, do ktorého používateľ zadá dve čísla a program vypíše všetky čísla medzi nimi. (Používateľ zadá dve čísla a nevieme ktoré je menšie a ktoré väčšie)