

PYTHON 3.6.1



Ing.Drgo Pavel,22.júna 2017,9:54
Modify,28.októbra 2017,sobota ,18:09

Ak bude čas na video

- [Ako začať programovať v PYTHONe](#)

[Andrej Blaho - Prečo učiť Python](#)

7 zákonov programovania

1. V každom programe je aspoň jedna chyba.
2. Každý program sa dá skrátit o aspoň jeden riadok.
3. Najjednoduchšie chyby sa najhoršie hľadajú.
4. Každou opravou chyby sa do programu zanesie nová chyba.
5. Keď sa Vám zdá, že program je v poriadku, určite ste niečo prehliadli.
6. Programátor robí čo vie, počítač si robí čo chce.
7. Program je tak inteligentný ako je inteligentný programátor

Úvodné informácie I.

- Python začal vytvárať Holanďan Guido van Rossum v roku 1989
- Názov dal jazyku podľa anglickej humoristickej skupiny Lietajúci cirkus Montyho Pythona
- Je multiplatformový(verzie pre Windows,Linux, Mac OS X)
- Python je vyvíjaný ako open-source, čo znamená, že si každý môže prečítať zdrojový kód
- Python – programovací jazyk pre výuku algoritmizácie a programovania
- Má jednoduchú a čistú syntax - k odsadzovanie sa používa tabulátor alebo medzery
- Python dobre spolupracuje s ostatnými programovacími jazykmi v ktorých sa dá využiť ako skriptovací jazyk

Úvodné informácie II.

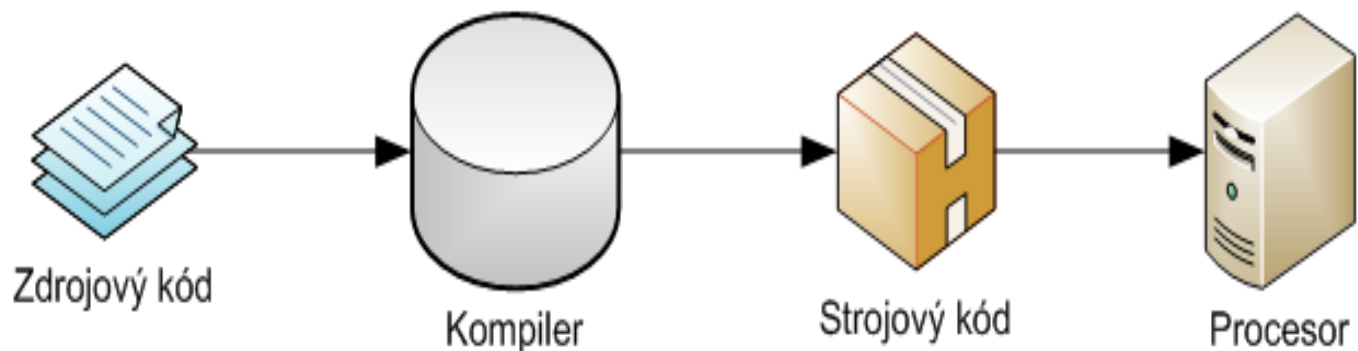
- Python má mnoho implementácií(PyPy, IronPython, Jython, Boost.Python)
- Používajú ho spoločnosti ako Google, Facebook, Mozilla, IBM, NASA, Instagram, YouTube, Intel
- jednoduchá **syntax** – je vhodný na učenie
- nie je nutná kompilácia kódu pred spustením
- dajú sa v ňom robiť aj **weby**
- v mnohých programoch sa používa na písanie skriptov
- **V Pythone je naprogramovaný BitTorrent** – slúži na sťahovanie torrentov
- Program spustíme v **IDLE (Python GUI)** – vývojové prostredie (Integrated Development Environment).

Úvodné informácie III.

- Na mnohých špičkových univerzitách sa učí ako úvodný jazyk, napr. Carnegie Mellon, Berkeley, Cornell, Caltech, Illinois, FMFI UK v Bratislave a Masarykovej Univerzite v Brne ...
- beží na rôznych platformách, napr. Windows, Linux, Mac.
- Na rozdiel od mnohých iných jazykov, ktoré sú kompilačné (napr. Pascal, C/C++) je Python interpretér.
- To znamená,že:
 - interpretér nevytvára spustiteľný kód (napr. .exe súbor vo Windows)
 - na spustenie programu musí byť v počítači nainštalovaný Python
 - interpretér umožňuje aj interaktívnu prácu s prostredím

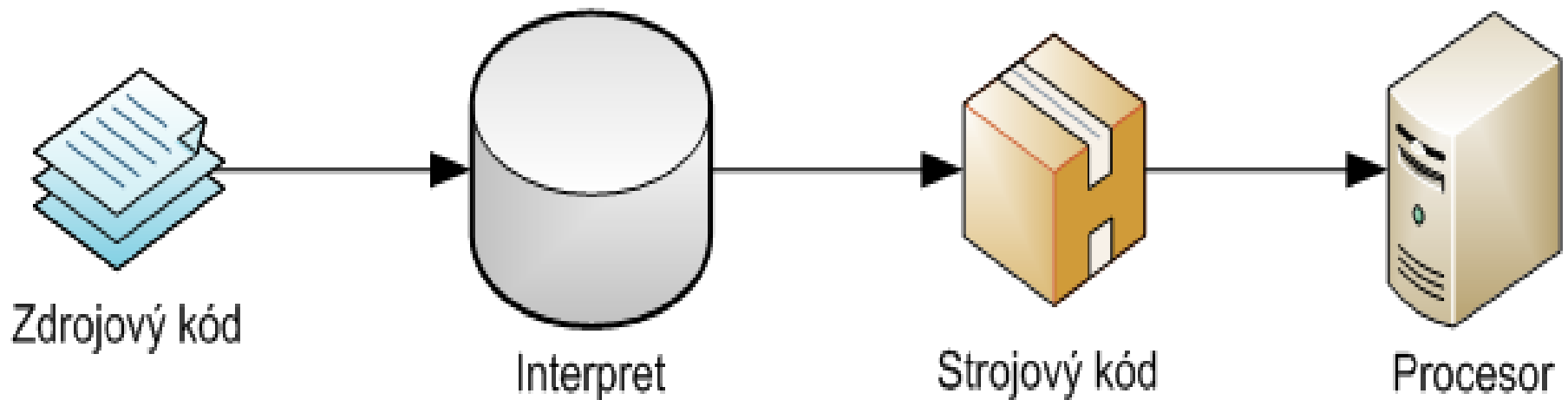
Kompilátor

- Kompilované (neriadené) jazyky majú svoj zdrojový kód v jazyku, ktorému ľudia dobre rozumejú.
- Tento zdrojový kód sa samozrejme musí preložiť do strojového kódu, aby ho bolo možné na procesore spustiť.
- Tento preklad zabezpečuje prekladač (kompiler), ktorý preloží naraz celý program do stroj. kódu.



INTERPRET

- Interpretácia sa snažia riešiť problém prenosnosti programov medzi rôznymi platformami
- tiež prichádza s vyšším komfortom pre programátora.
- Interpret neprekladá program celý naraz, ale prekladá len to, čo je v danej chvíli potrebuje.



Výhody interpreta

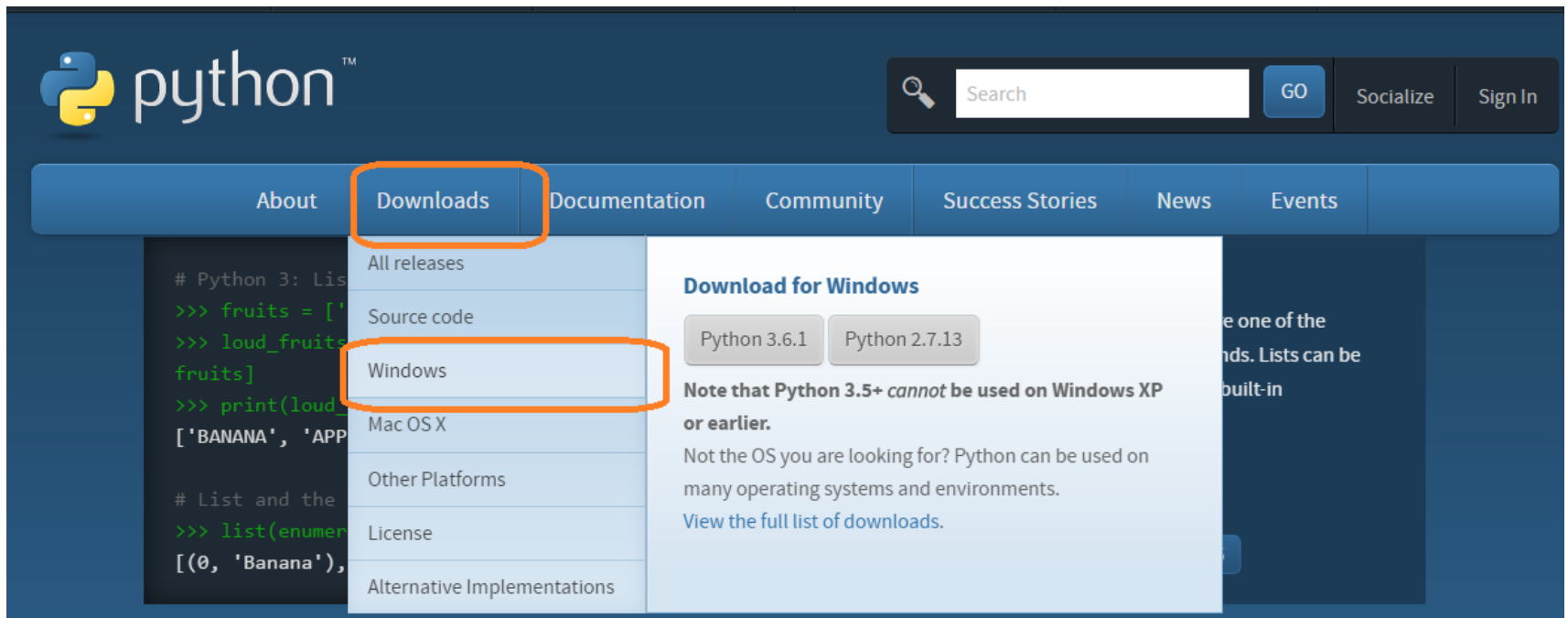
- **Prenositeľnosť** : Program je plne prenositeľný, ak existuje interpret pre danú platformu, pôjde tam zdrojový kód programu spustiť (a vývoj interpretu je ľahšie, než vývoj kompilátora).
- **Jednoduchšie vývoj** - Vo vyšších jazykoch robí správu pamäte, tzv. Garbage collector. Často tiež nemusíme ani zadávať dátové typy a máme k dispozícii vysoko komfortné kolekcie a ďalšie štruktúry.
- **Stabilita** - Vďaka tomu, že interpret kódu rozumie, predíde chybám, ktoré by skompilovaný program inak pokojne vykonal.
- **Jednoduchá editácia** - Program môžeme vyvíjať po častiach a nahrávať na cieľové umiestnenie, vďaka tomu, že sa nemusí kompilovať, ho je možné jednoducho editovať "za behu".

Interpreter má tri zásadné nevýhody :

- **Rýchlosť** - Interpretácia môže byť mnohokrát veľmi pomalá a program tak plne nevyužíva výkon počítača.
- **Často ťažké hľadanie chýb** - Vďaka kompilácii za behu sa chyby v kóde objaví až v tú chvíľu, kedy je kód spustený. To môže byť niekedy veľmi nepríjemné.
- **Zraniteľnosť** - Pretože sa program šíri v podobe zdrojového kódu, každý do neho môže zasahovať alebo kradnúť jeho časti.

Inštalácia Pythonu 1.krok

- zo stránky: <https://www.python.org/>



The screenshot shows the Python.org website interface. The Python logo is in the top left. A search bar with a magnifying glass icon and a 'GO' button is in the top right. Below the search bar is a navigation menu with the following items: About, Downloads, Documentation, Community, Success Stories, News, and Events. The 'Downloads' menu is open, showing a list of options: All releases, Source code, Windows, Mac OS X, Other Platforms, License, and Alternative Implementations. The 'Windows' option is highlighted with an orange box. To the right of the 'Downloads' menu, there is a 'Download for Windows' section. It features two buttons: 'Python 3.6.1' and 'Python 2.7.13'. Below these buttons, there is a note: 'Note that Python 3.5+ cannot be used on Windows XP or earlier.' and a link: 'View the full list of downloads.'

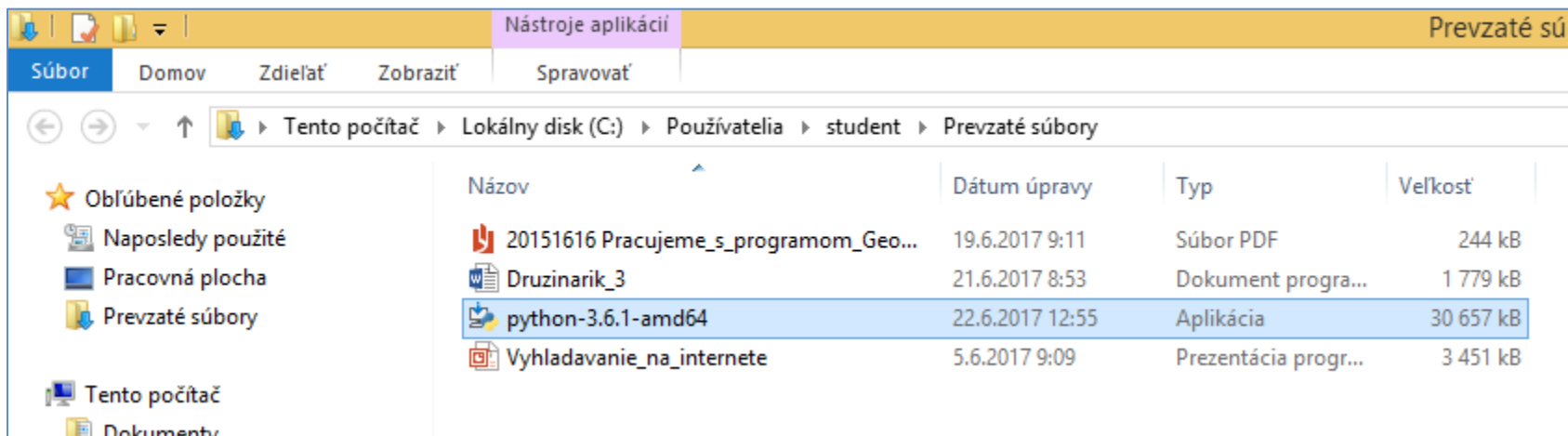
```
# Python 3: Lis
>>> fruits = ['
>>> loud_fruits
fruits]
>>> print(loud_
['BANANA', 'APP
# List and the
>>> list(enumer
[(0, 'Banana'),
```

Inštalácia Pythonu 2.krok

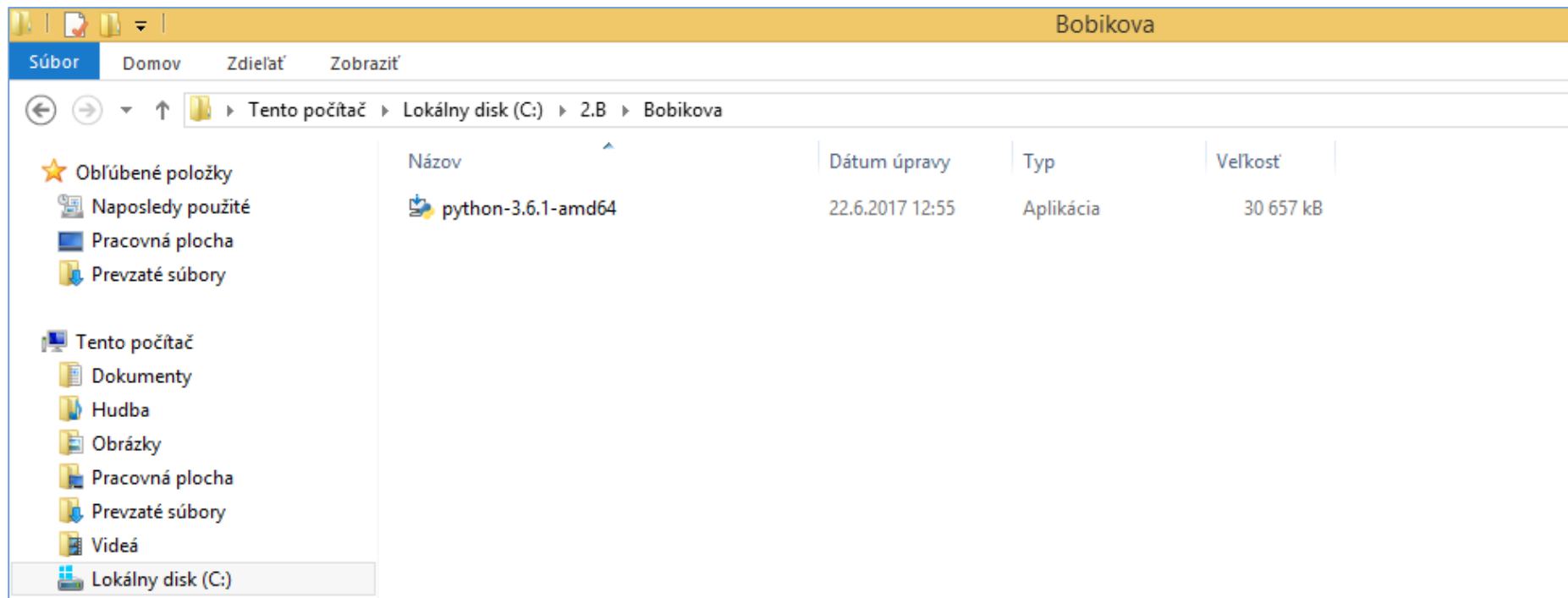
Python Releases for Windows

- [Latest Python 2 Release - Python 2.7.13](#)
- [Latest Python 3 Release - Python 3.6.1](#)
- [Python 3.6.2rc1 - 2017-06-17](#)
 - [Download Windows x86 web-based installer](#)
 - [Download Windows x86 executable installer](#)
 - [Download Windows x86 embeddable zip file](#)
 - [Download Windows x86-64 web-based installer](#)
 - [Download Windows x86-64 executable installer](#)
 - [Download Windows x86-64 embeddable zip file](#)
 - [Download Windows help file](#)
- [Python 3.6.1 - 2017-03-21](#)
 - [Download Windows x86 web-based installer](#)
 - [Download Windows x86 executable installer](#)
 - [Download Windows x86 embeddable zip file](#)
 - [Download Windows x86-64 web-based installer](#)
 - [Download Windows x86-64 executable installer](#)

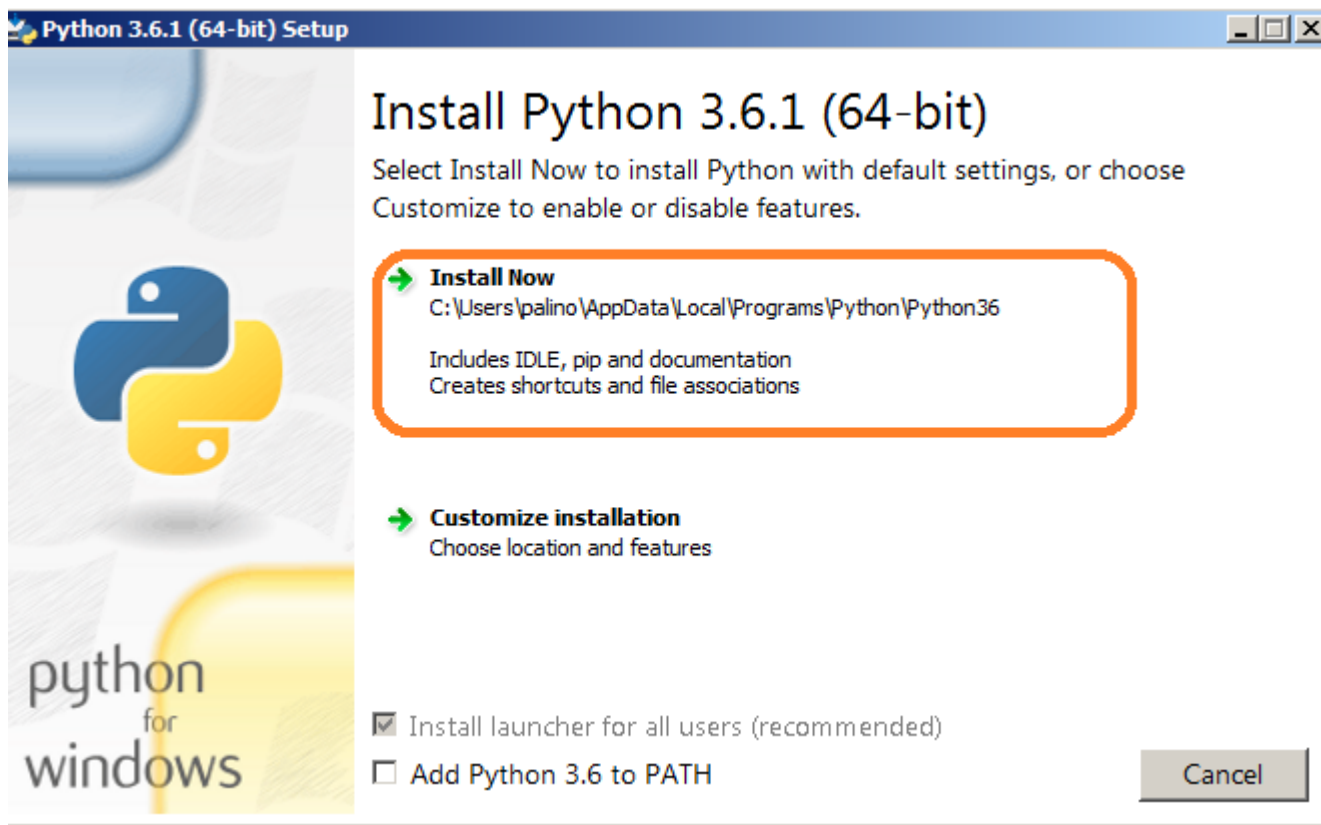
Inštalácia Pythonu 3.krok



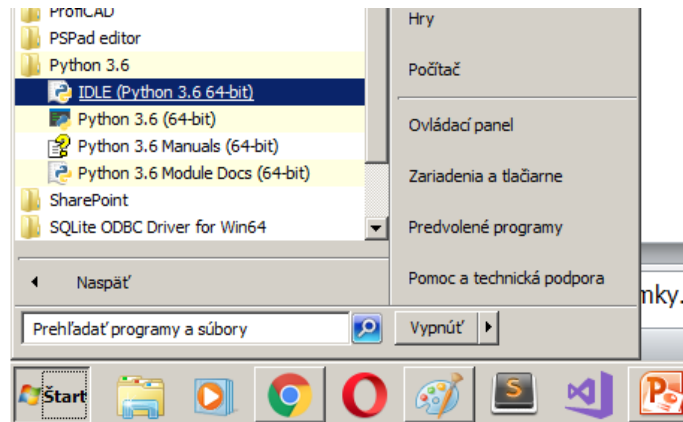
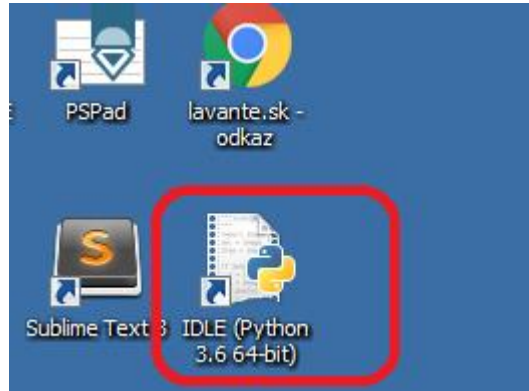
Inštalácia Pythonu 4.krok



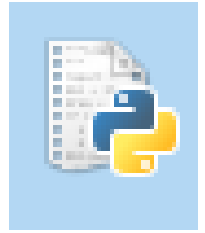
Inštalácia Pythonu 5.krok



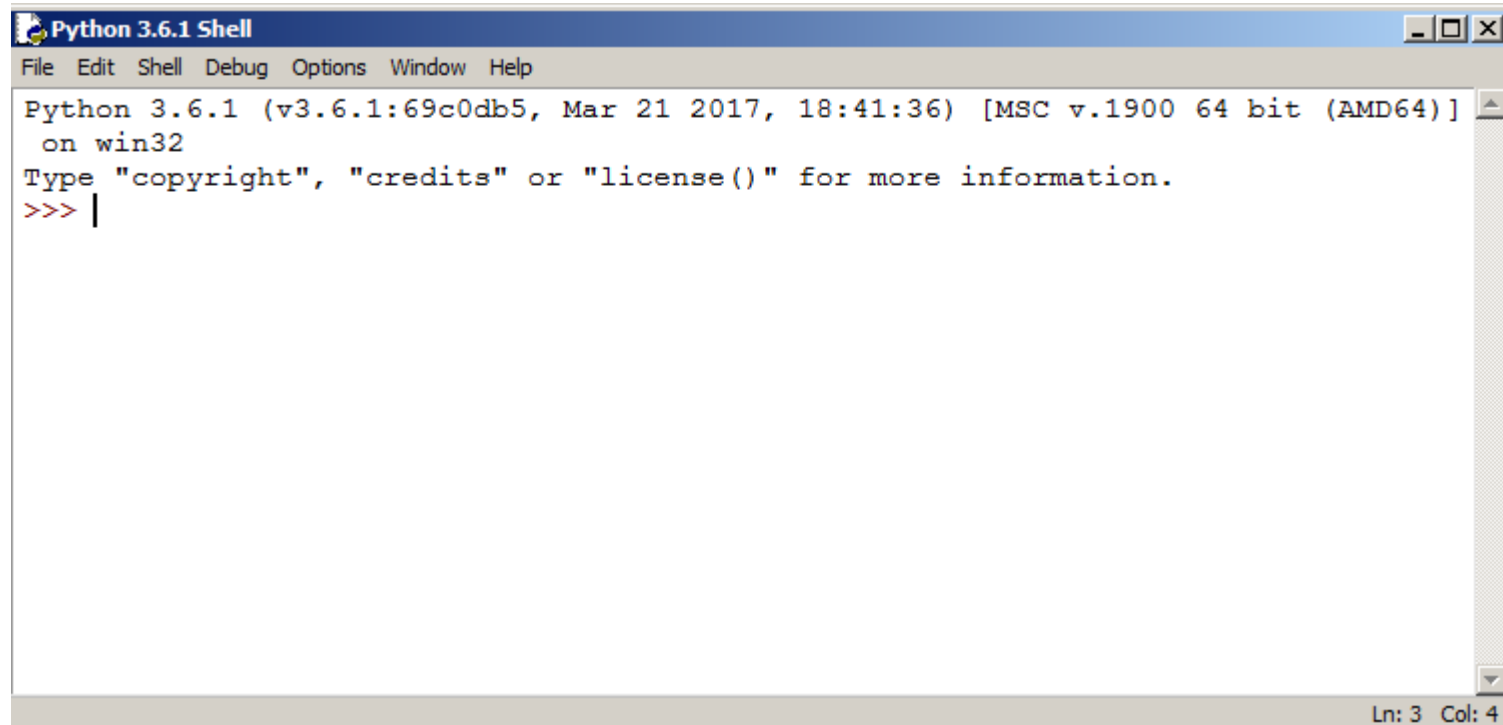
Spustenie Pythonu



Úvod do Pythonu a IDLE (Integrated Development Environment).



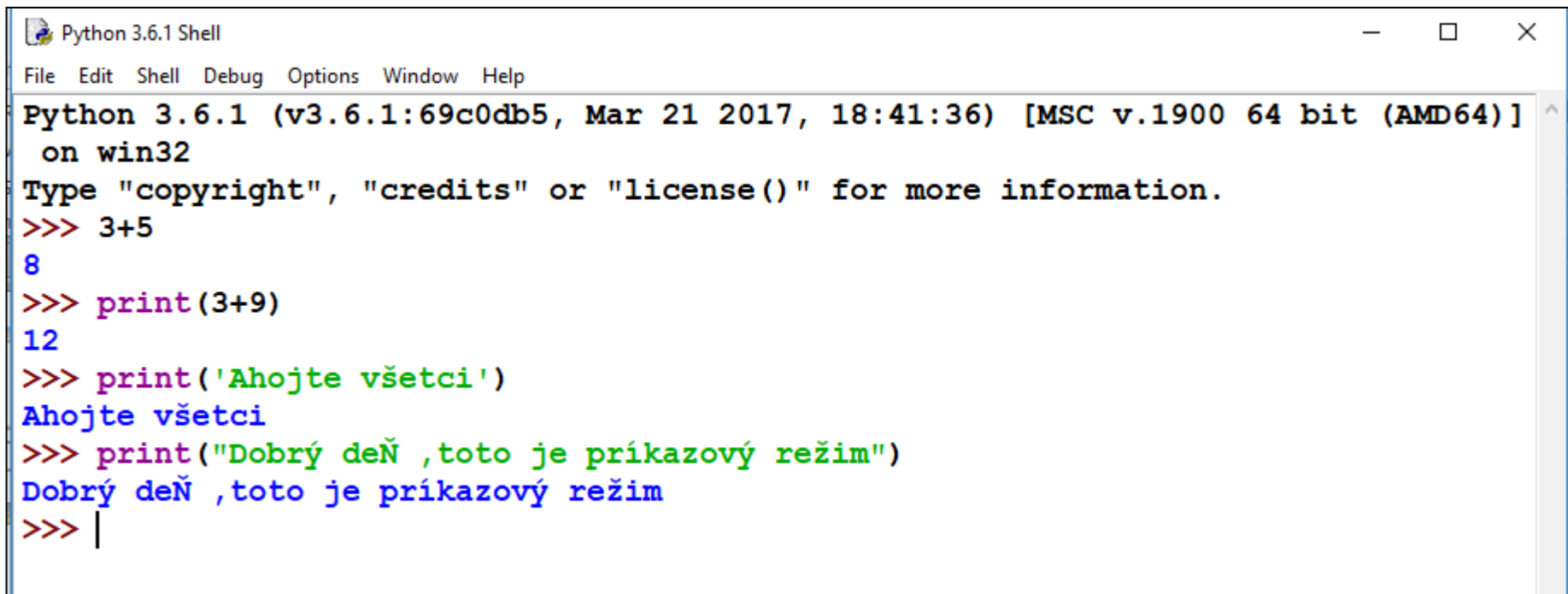
- Po spustení prostredia sa objaví jedno okno, tzv. Shell, v ktorom môžeme písať priamo príkazy za znaky `>>>`.
- Takéto programovanie budeme nazývať príkazový režim.
- **Príkazový režim:** Za šípkami `>>>` píšeme príkazy a stlačením tlačidla ENTER sa príkazy vykonajú.

A screenshot of a Python 3.6.1 Shell window. The title bar reads "Python 3.6.1 Shell". The menu bar includes "File", "Edit", "Shell", "Debug", "Options", "Window", and "Help". The main text area shows the following output:

```
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>> |
```

The status bar at the bottom right indicates "Ln: 3 Col: 4".

Príkazový režim

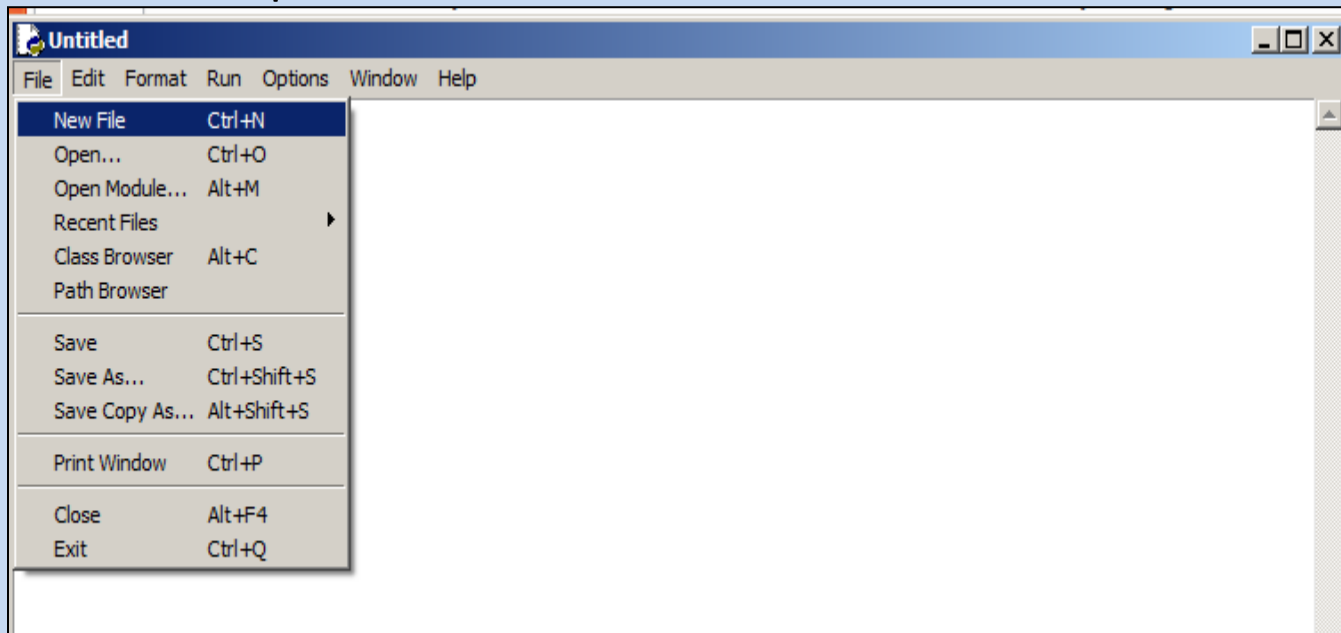


```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>> 3+5
8
>>> print(3+9)
12
>>> print('Ahojte všetci')
Ahojte všetci
>>> print("Dobrý deň ,toto je príkazový režim")
Dobrý deň ,toto je príkazový režim
>>> |
```

VYSKÚŠAJTE SI TO!!!!!!!!!!

Programovací režim- 1.možnosť ako vytvoriť súbor

1. V okne shell z hlavného menu File vyberieme menu File a v ňom ponuku **New File**.

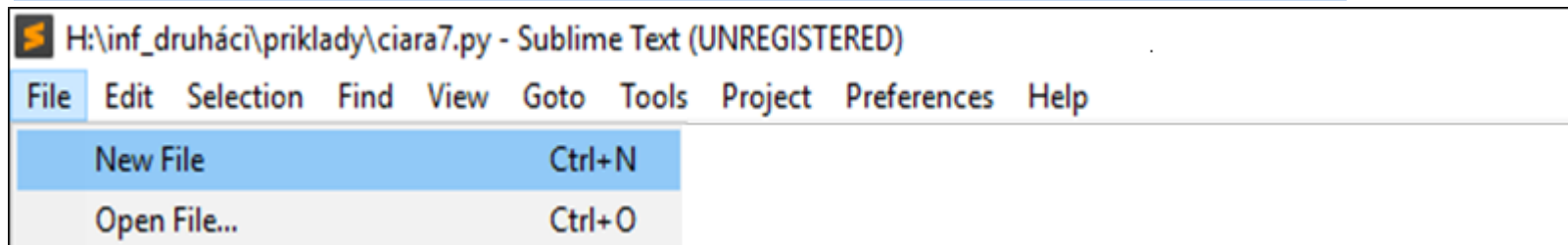


2. Otvorilo sa nám nové okno, kde zapíšeme program.
3. Program uložíme s príponou menoSuboru.py

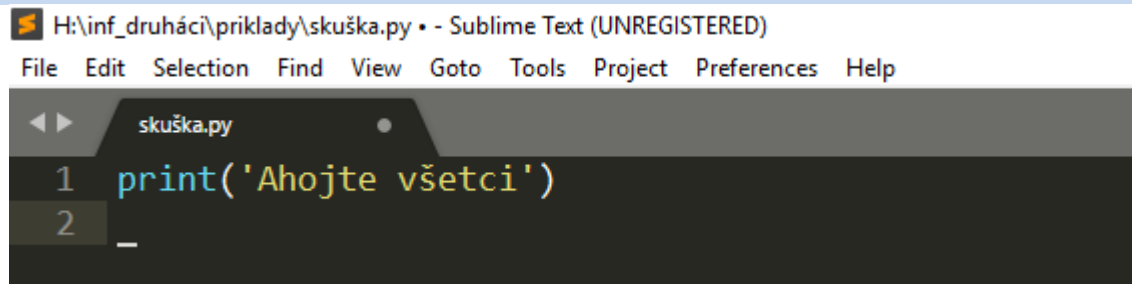
Programovací režim- 2.možnosť ako vytvoriť súbor

1.Spustiť program SUBLIME TEXT3

2.File→New File



3.Napísať program

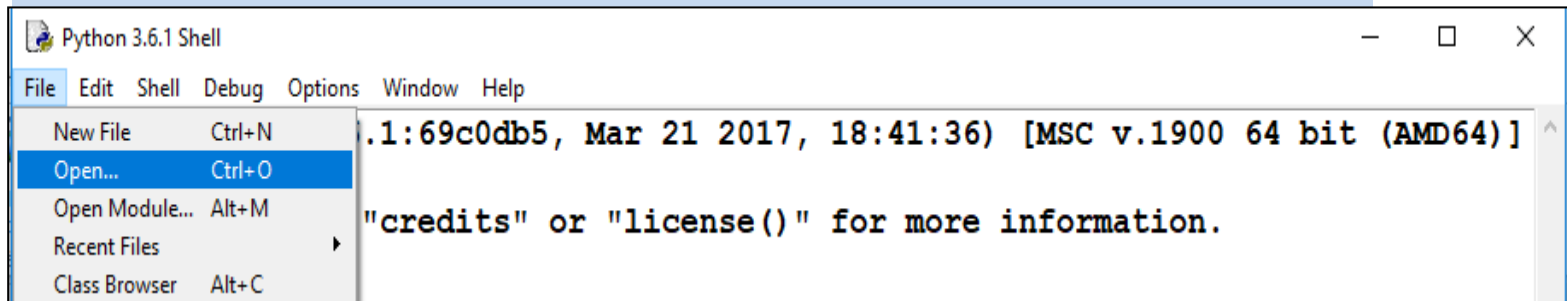


4.Uložiť súbor s príponou .py

Programovací režim- Ako otvoriť súbor

1.Spustiť program PYTHON shell

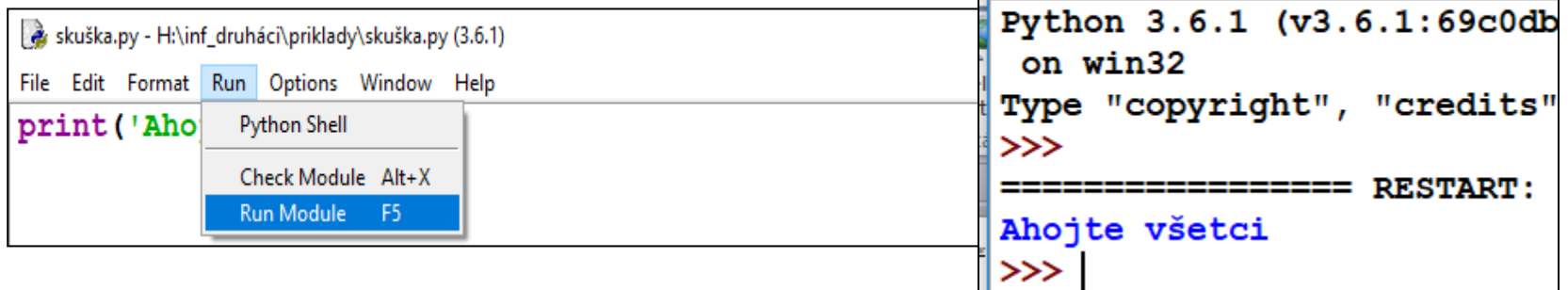
2.File→Open..



4.Zobrazí sa otvorený súbor

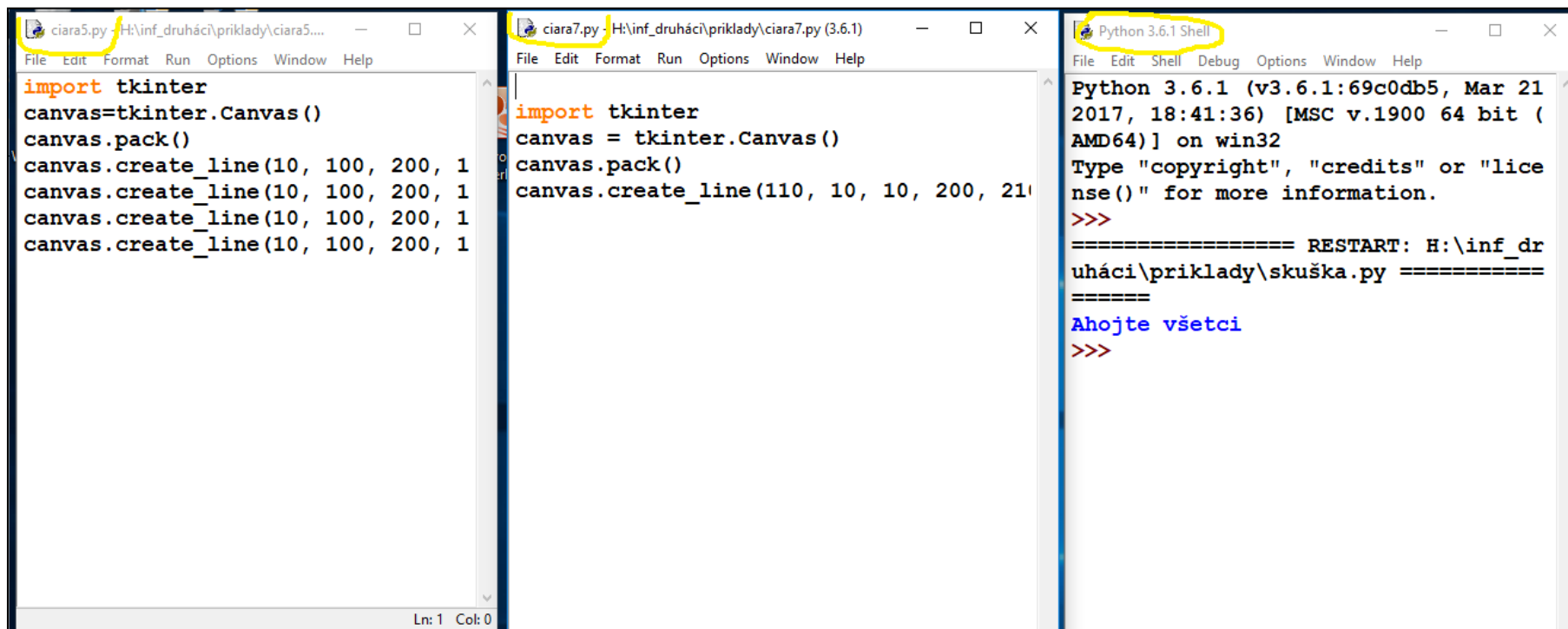


5.Spustíme otvorený súbor



Otvorené okná

Upravíme si veľkosť všetkých otvorených okien, tak aby sme ich mali dostupné



The screenshot shows three overlapping windows. The leftmost window is titled 'ciara5.py' and contains the following Python code:

```
import tkinter
canvas=tkinter.Canvas()
canvas.pack()
canvas.create_line(10, 100, 200, 1
canvas.create_line(10, 100, 200, 1
canvas.create_line(10, 100, 200, 1
canvas.create_line(10, 100, 200, 1
```

The middle window is titled 'ciara7.py' and contains the following Python code:

```
import tkinter
canvas = tkinter.Canvas()
canvas.pack()
canvas.create_line(110, 10, 10, 200, 21
```

The rightmost window is titled 'Python 3.6.1 Shell' and shows the output of a Python script:

```
Python 3.6.1 (v3.6.1:69c0db5, Mar 21
2017, 18:41:36) [MSC v.1900 64 bit (
AMD64)] on win32
Type "copyright", "credits" or "lice
nse()" for more information.
>>>
===== RESTART: H:\inf_dr
uháci\priklady\skuska.py =====
=====
Ahojte všetci
>>>
```

Aktívne je to okno do ktorého klikneme

výstupná funkcia(príkaz)

- **print()**
- Je to volanie špeciálnej funkcie
- táto funkcia vypisuje hodnoty výrazov, ktoré sú uvedené medzi zátvorkami
- hodnoty sú pri výpise oddelené medzerami
- `print()` bez parametrov spôsobí len zariadkovanie výpisu, teda vloží na momentálne miesto prázdny riadok
- Ak znakový ret'azec obsahuje dvojicu znakov `'\n'`, tieto označujú, že pri výpise funkciou `print()` prejde na nový riadok.

vstupná funkcia(príkaz)

- `input()`
- je tiež funkcia, ktorá najprv vypíše zadaný znakový ret'azec (ak je zadaný)
- a potom čaká na vstupný ret'azec ukončený ENTER
- funkcia vráti tento zadaný ret'azec
- Vstupná informácia má typ str-je znaková(nie číselná)

Znakové reťazce

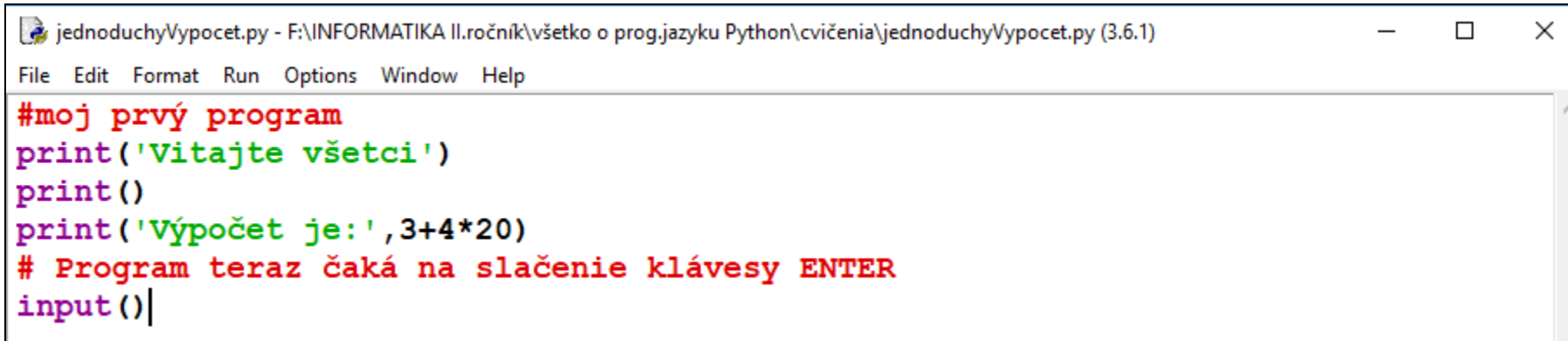
- ich dĺžka (počet znakov) je obmedzená len kapacitou pracovnej pamäte Pythonu
- text uzatvárame do apostrofov 'text' alebo úvodzoviek
- – oba zápisy sú rovnocenné - reťazec musí končiť tým istým znakom ako začal (apostrof alebo úvodzovka)
- – takto zadaný reťazec nesmie presiahnuť jeden riadok
- môže obsahovať aj písmená s diakritikou
- prázdny reťazec má dĺžku 0 a zapisujeme ho ako ""

Příklad1.jednoduchyVypocer.py

```
#AltGr+x  
#moj prvý program  
print('Vitajte všetci')  
print()  
print('Výpočet je:',3+4*20)  
# Program teraz čaká na stlačenie klávesy  
ENTER  
input('Stlač klávesu ENTER')
```

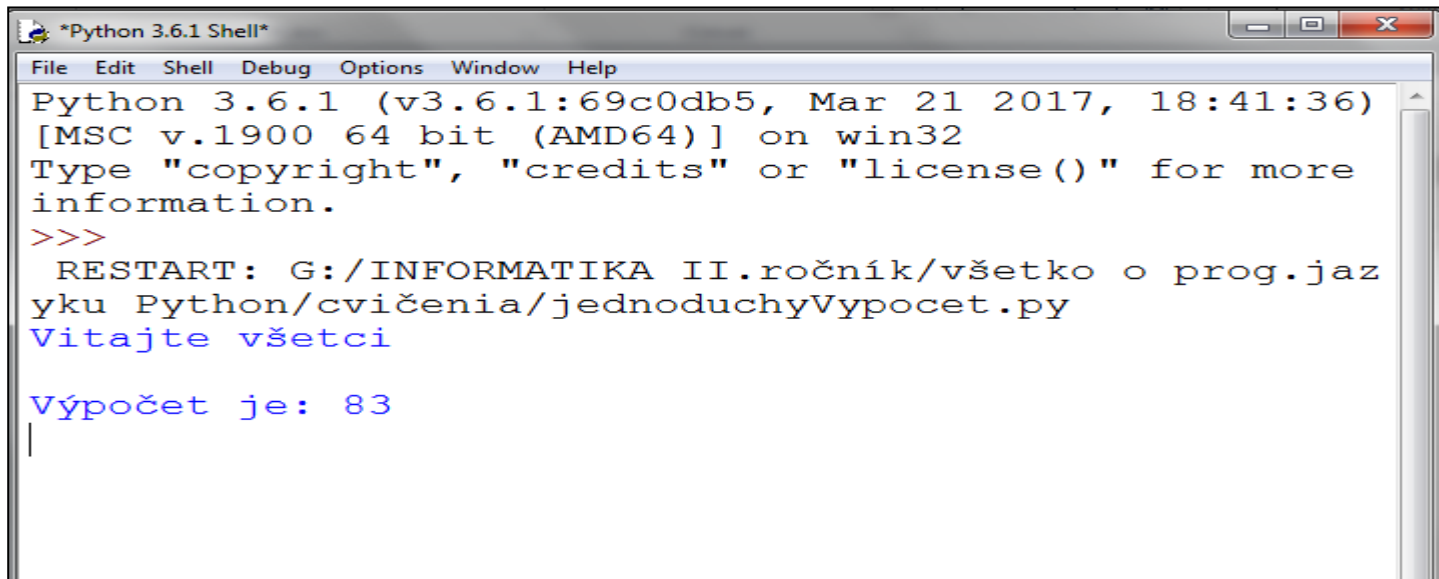
Shell

Spustenie uloženého programu.



The screenshot shows a window titled "jednoduchyVypocet.py - F:\INFORMATIKA II.ročník\všetko o prog.jazyku Python\cvičenia\jednoduchyVypocet.py (3.6.1)". The menu bar includes File, Edit, Format, Run, Options, Window, and Help. The code in the editor is as follows:

```
#moj prvý program
print('Vitajte všetci')
print()
print('Výpočet je:', 3+4*20)
# Program teraz čaká na stlačenie klávesy ENTER
input()
```

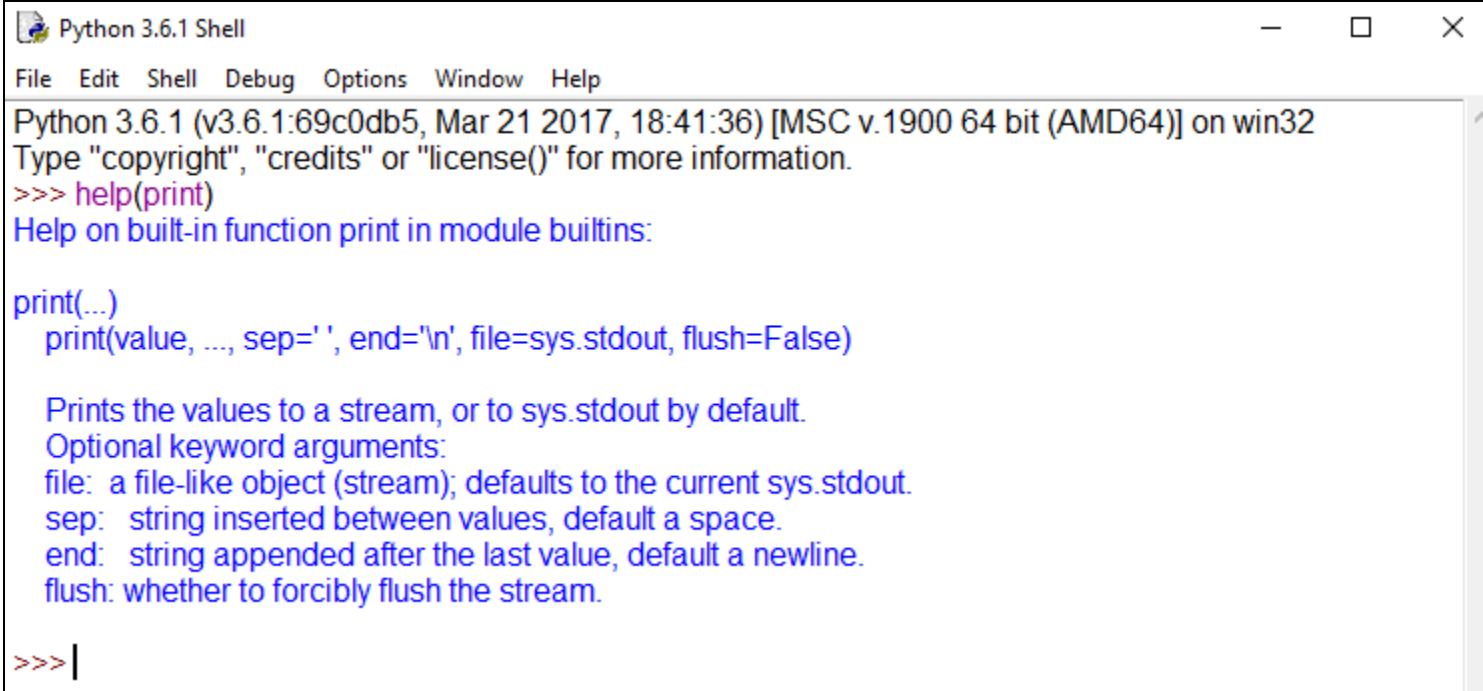


The screenshot shows a "Python 3.6.1 Shell" window with a menu bar (File, Edit, Shell, Debug, Options, Window, Help). The output of the script is displayed as follows:

```
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36)
[MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more
information.
>>>
  RESTART: G:/INFORMATIKA II.ročník/všetko o prog.jaz
yku Python/cvičenia/jednoduchyVypocet.py
  Vitajte všetci

  Výpočet je: 83
  |
```

Použitie funkcie help()-nápoveda



```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> help(print)
Help on built-in function print in module builtins:

print(...)
    print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)

    Prints the values to a stream, or to sys.stdout by default.
    Optional keyword arguments:
    file: a file-like object (stream); defaults to the current sys.stdout.
    sep: string inserted between values, default a space.
    end: string appended after the last value, default a newline.
    flush: whether to forcibly flush the stream.

>>> |
```

Typy údajov

- Hodnoty (konštanty alebo výrazy) môžu byť rôznych typov.
- V Pythone má každý typ svoje meno:
- **int** ako celé čísla, napr. 0, 1, 15, -123456789, ...
- **float** ako desatinné čísla, napr. 0.0, 3.14159, 2.00000000001, 33e50, ...
- **str** ako znakové reťazce, napr. 'a', "äbc", " I'm happy"
- Typ ľubovol'nej hodnoty vieme v Pythone zistiť pomocou štandardnej funkcie **type()**.

Celočíselné operácie s číslami

oba operandy musia byť celočíselného typu

+ súčet, napr. $1 + 2$ má hodnotu 3

- rozdiel, napr. $2 - 5$ má hodnotu -3

* násobenie, napr. $3 * 37$ má hodnotu 111

/ delenie ,napr. $21/7$ dá hodnotu 3.0

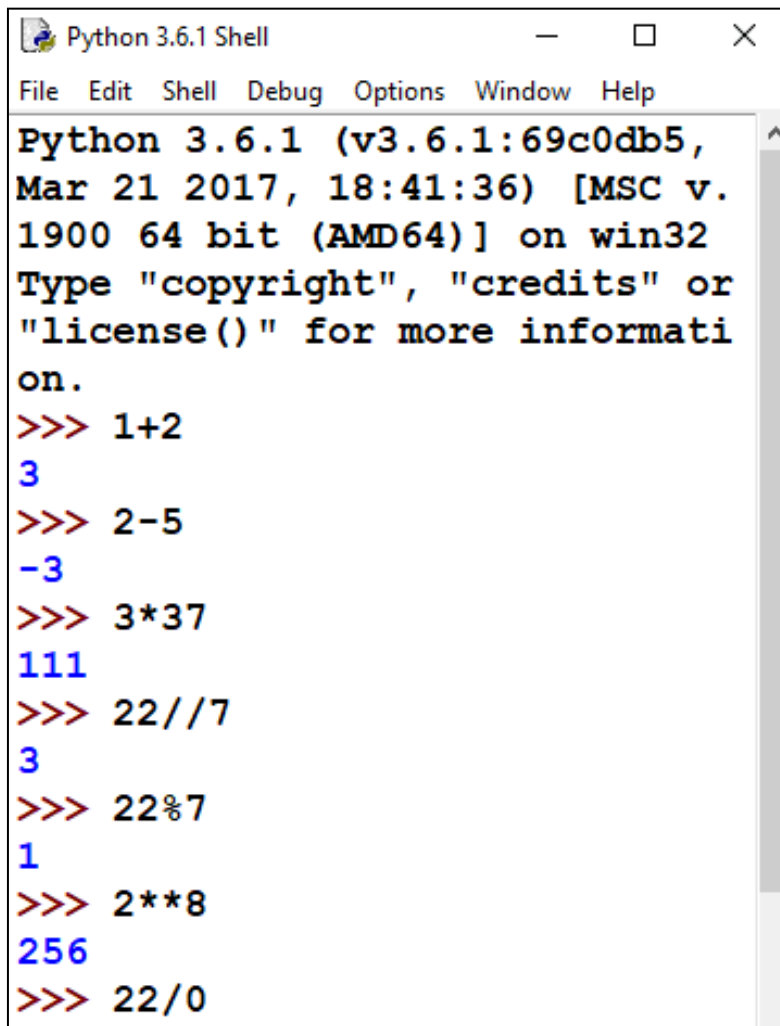
// celočíselné delenie, napr. $22 // 7$ má hodnotu 3

% zvyšok po delení, napr. $22 \% 7$ má hodnotu 1

** umocňovanie, napr. $2 ** 8$ má hodnotu 256

zrejme nemôžeme deliť 0

Skúšanie celočíselných operácií



```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5,
Mar 21 2017, 18:41:36) [MSC v.
1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or
"license()" for more informati
on.
>>> 1+2
3
>>> 2-5
-3
>>> 3*37
111
>>> 22//7
3
>>> 22%7
1
>>> 2**8
256
>>> 22/0
```

operácie s desatinnými číslami

aspoň jeden operand musí byť desatinného typu (okrem delenia /)

- + súčet, napr. $1 + 0.2$ má hodnotu 1.2
- rozdiel, napr. $6 - 2.86$ má hodnotu 3.14
- * násobenie, napr. $1.5 * 2.5$ má hodnotu 3.75
- / delenie, napr. $23 / 3$ má hodnotu 7.666666666666667
- // delenie zaokrúhlené nadol, napr. $23.0 // 3$ má hodnotu 7.0
- % zvyšok po delení, napr. $23.0 \% 3$ má hodnotu 2.0
- ** umocňovanie, napr. $3 ** 3$ má hodnotu 27.0
zrejme nemôžeme deliť 0

Skúšanie operácií s pohyblivou čiarkou

```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 2
1 2017, 18:41:36) [MSC v.1900 64 bi
t (AMD64)] on win32
Type "copyright", "credits" or "lic
ense()" for more information.
>>> 1+0.2
1.2
>>> 6-2.86
3.14
>>> 1.5*2.5
3.75
>>> 23/3
7.666666666666667
>>> 23.0//3
7.0
>>> 23.0%3
2.0
>>> 3**3
27
>>> 27//0
Traceback (most recent call last):
  File "<pyshell#7>", line 1, in <m
```

operácie so znakovými reťazcami

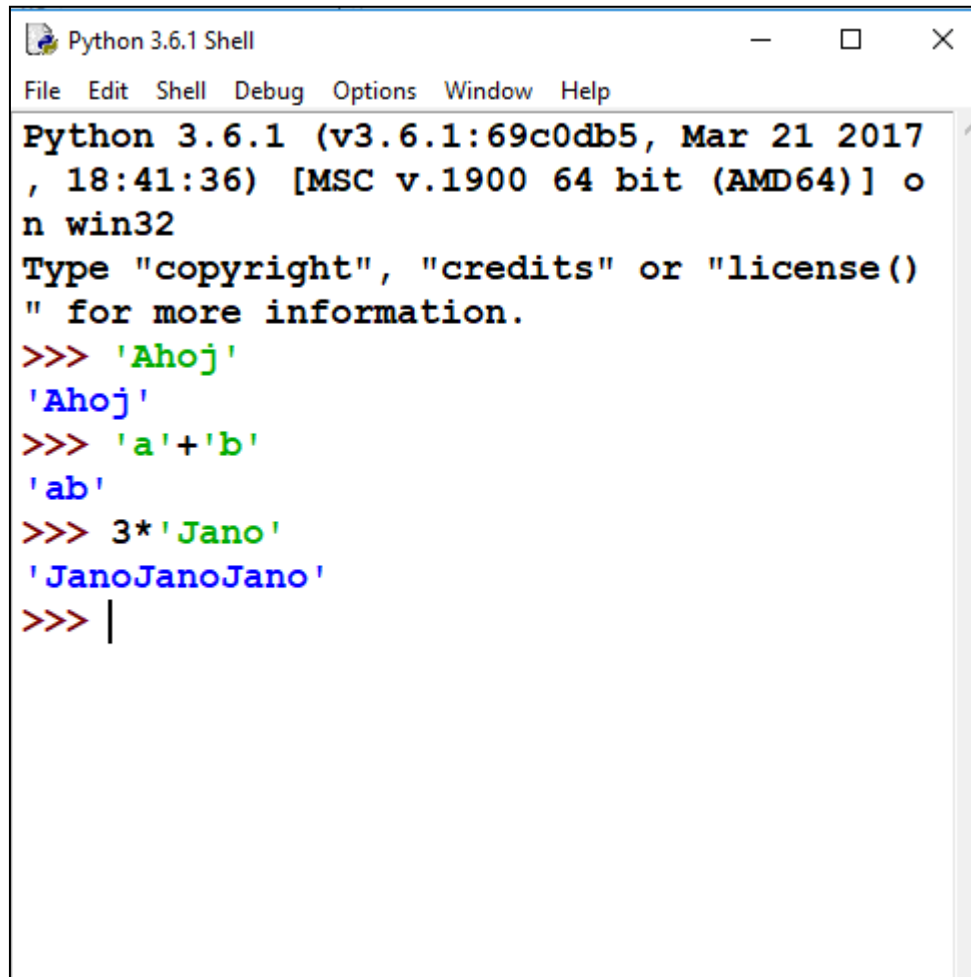
+ zretáženie (spojenie dvoch reťazcov),
napr. 'a' + 'b' má hodnotu 'ab'

* viacnásobné zretáženie toho istého
reťazca, napr. 3 * 'x' má hodnotu 'xxx',

Viacnásobné zretáženie je iba u Pythona

Znak apostrof je: `AltGr+p` v interaktívnom
režime

Operácie so znakovými reťazcami



```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017
, 18:41:36) [MSC v.1900 64 bit (AMD64)] o
n win32
Type "copyright", "credits" or "license()"
" for more information.
>>> 'Ahoj'
'Ahoj'
>>> 'a'+ 'b'
'ab'
>>> 3* 'Jano'
'JanoJanoJano'
>>> |
```

Premenné a priradenie

Premenná je miesto v pamäti počítača vyhradené na ukladanie obsahov premenných

Meno premennej:

- môže obsahovať písmená, číslice a znak podčiariť
- pozor na to, že v Pythone sa rozlišujú malé a veľké písmená
- musí sa líšiť od Pythonovských príkazov (napr. for, if, ...)

Premenná sa vytvorí priradovacím príkazom :

premenná = hodnota

tento zápis znamená, že do premennej sa má priradiť zadaná hodnota

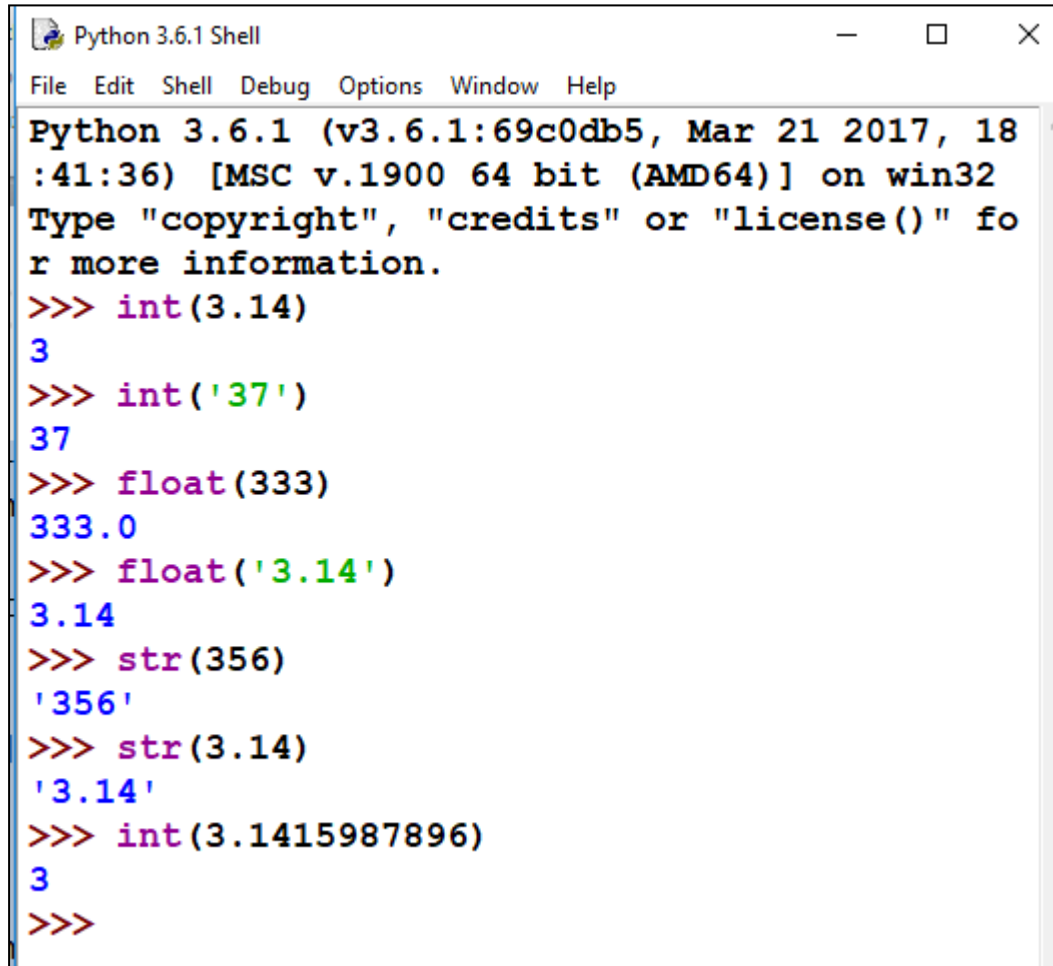
V Pythone majú programátori dohodu, že na premenné sa používajú len malé písmená (prípadne aj číslice a podčiarkovník).

PARSOVANIE-pretypovanie hodnôt

Mená typov `int`, `float` a `str` zároveň slúžia ako mená pretypovacích funkcií, ktoré dokážu z jedného typu vyrobiť hodnotu iného typu:

- `int(hodnota)` -z danej hodnoty vyrobí celé číslo, napr.
 - `int(3.14) => 3`
 - `int('37') => 37`
- `float(hodnota)` -z danej hodnoty vyrobí desatinné číslo, napr.
 - `float(333) => 333.0`
 - `float('3.14') => 3.14`
- `str(hodnota)` z danej hodnoty vyrobí znakový reťazec, napr.
 - `str(356) => '356'`
 - `str(3.14) => '3.14'`

PARSOVANIE-pretypovanie hodnôt



```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18
:41:36) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" fo
r more information.
>>> int(3.14)
3
>>> int('37')
37
>>> float(333)
333.0
>>> float('3.14')
3.14
>>> str(356)
'356'
>>> str(3.14)
'3.14'
>>> int(3.1415987896)
3
>>>
```

PARSOVANIE- príkazový režim

- `input()` zadanú hodnotu vráti nie ako číslo, ale ako znakový reťazec, teda nie 1.2 ale '1.2'.
- Najčastejšie potrebujeme ale číslo 1.2 pre matematické operácie
- Potrebujeme takýto reťazec prekonvertovať na číslo
- Na to použijeme funkciu `int()`

Vyskúšajte si cvičenie na ďalšom slide

Treba mať nastavené prostredie Shell-interaktívny režim

Python 3.6.1 Shell

File Edit Shell Debug Options Window Help

Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit
on win32

Type "copyright", "credits" or "license()" for more information.

>>> # Na ďalší riadok napíš toto a stlač ENTER

>>> a = input('Zadaj prvé číslo súčtu a stlač ENTER...' + '\n')

Zadaj prvé číslo súčtu a stlač ENTER...

21

>>> # Na ďalší riadok napíš toto a stlač ENTER

>>> b = input('Zadaj druhé číslo súčtu a stlač ENTER...' + '\n')

Zadaj druhé číslo súčtu a stlač ENTER...

2

>>> # Na ďalší riadok napíš toto a stlač ENTER

>>> print(a+b)

212

>>> # Toto som nechcel, 212 nie je správny výsledok

>>> |

212

```
>>> # Toto som nechcel, 212 nie je správny výsledok
>>> # Musíme pretypovať a a b
>>> # Toto napíš a stlač ENTER
>>> a = int(a)
>>> b = int(b)
>>> c = a + b
>>> # toto napíš
>>> print('Výsledok je'+ c)
Traceback (most recent call last):
  File "<pysshell#13>", line 1, in <module>
    print('Výsledok je'+ c)
TypeError: must be str, not int
>>> # opravíme to
>>> print('Výsledok je'+ str(c))
Výsledok je23
>>> # To je už správne
>>>
```

ZHRNUTIE

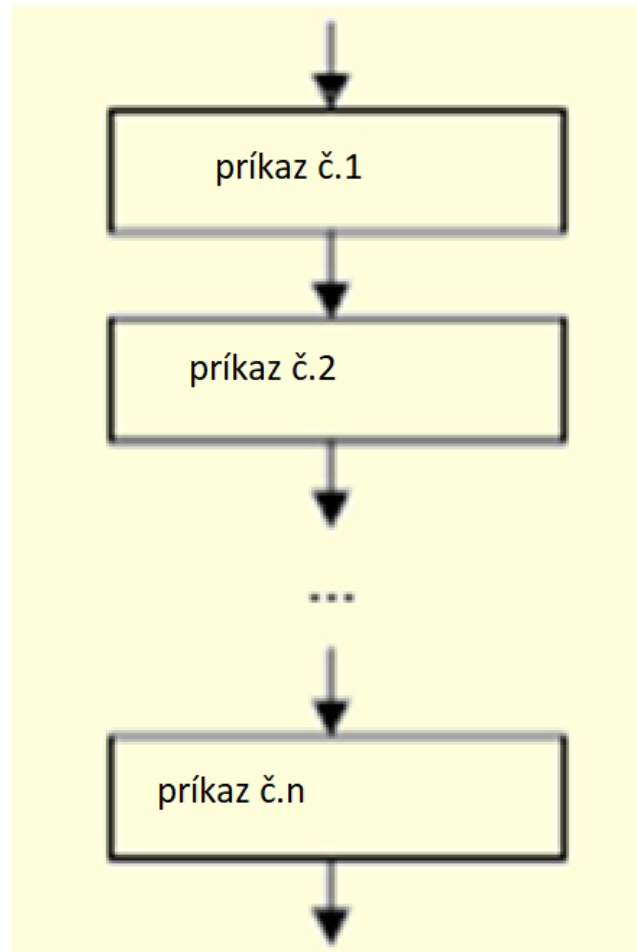
- príkazový režim -za výzvu (prompt) `>>>` zadávame výrazy a príkazy
- programovací režim- skripty vytvárame v textovom súbore (prípona `.py`) a spúšťame pomocou klávesy `<F5>`
- funkcia `print()` vypisuje zadané hodnoty
- funkcia `input()` čaká na stlačenie ENTER, funkcia vráti používateľom zadaný reťazec
- typy hodnôt a základné typy premenných sú: `int`, `float`, `str`,
- funkcia `type(print)` slúži ako help pre funkciu `print`
- premenná obsahuje referenciu na hodnotu,
- vznikne pomocou prirad'ovacieho príkazu

Písomka

- 1.Charakteristika programovacieho jazyka Python
- 2.Vymenujte režimy práce Pythonu
- 3.Aký rozdiel je medzi príkazom print a input
- 4.Vymenujte typy údajov
- 5.Čo je to parsovanie údajov
- 6.Aká je táto operácia: %

body	známka
6	1
5	2
4	3
3	4
2,1,0	5

Příklady na sekvenciu



Klávesové skratky

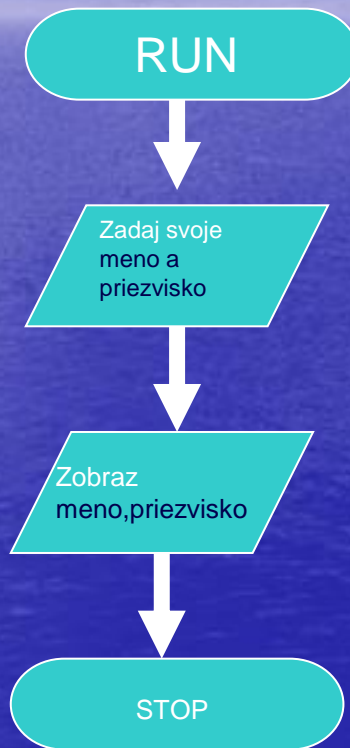
- Pravý Alt+p.....apostrof (‘)
- Shift +ô.....úvodzovky(“ ”)
- Pravý Alt+ô.....dolar (\$)
- Pravý Alt+x.....hashtag (#)
- Ľavý alt+92..... backslash(\),a EN klav.ň



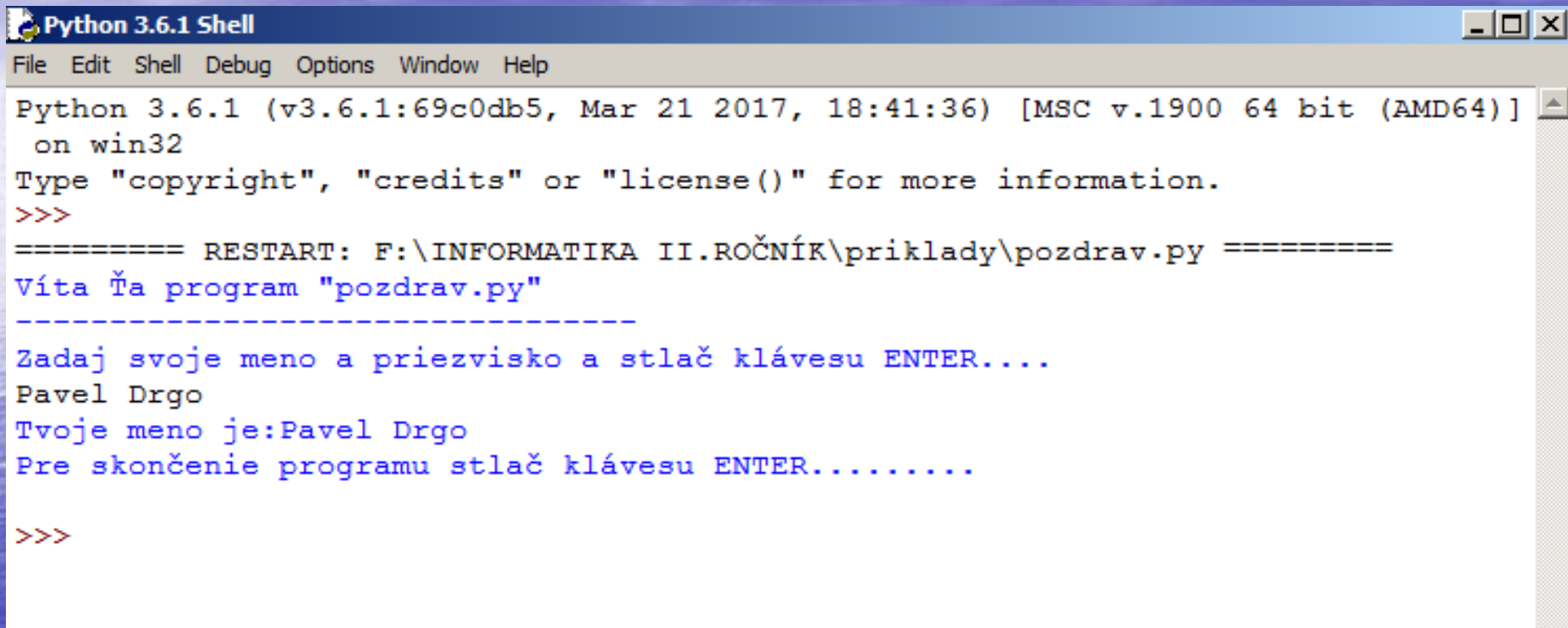
POZDRAV

Príklady na algoritmickú konštrukciu Sekvencia

Pr.1.Zostavte algoritmus na zadanie a zobrazenie mena a priezviska(textu)



ZADANIE-pozdrav.py



```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: F:\INFORMATIKA II.ROČNÍK\priklady\pozdrav.py =====
Víta Ťa program "pozdrav.py"
-----
Zadaj svoje meno a priezvisko a stlač klávesu ENTER....
Pavel Drgo
Tvoje meno je:Pavel Drgo
Pre skončenie programu stlač klávesu ENTER.....
>>>
```

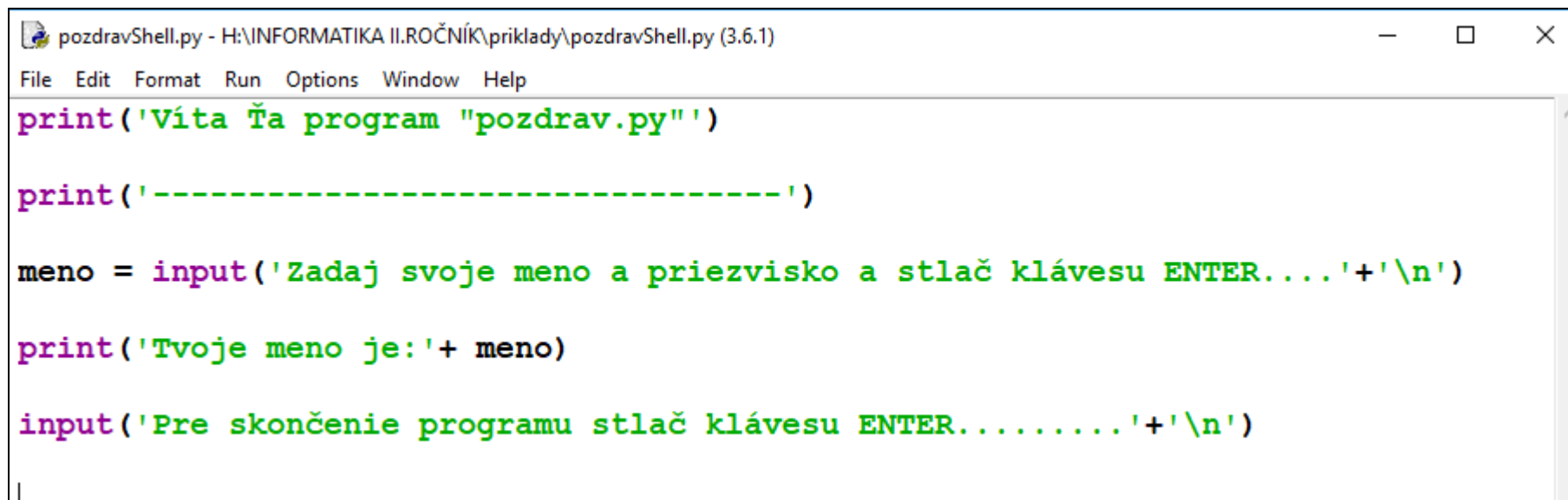

- 1) `print('Víta Ťa program "pozdrav.py"')`
- 2) `print('-----')`
- 3) `meno = input('Zadaj svoje meno a priezvisko a stlač klávesu ENTER....'+'\n')`
- 4) `print('Tvoje meno je:'+ meno)`
- 5) `input('Pre skončenie programu stlač klávesu ENTER.....'+'\n')`

LADENIE PROGRAMU

```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>> print('Víta Ěa program "pozdrav.py"')
Víta Ěa program "pozdrav.py"
>>> print('-----')
-----
>>> meno = input('Zadaj svoje meno a priezvisko a stlaĉ klávesu ENTER....'+'\n')
Zadaj svoje meno a priezvisko a stlaĉ klávesu ENTER....
Ing.Drgo Pavel
>>> print('Tvoje meno je:'+ meno)
Tvoje meno je:Ing.Drgo Pavel
>>> input('Pre skonĉenie programu stlaĉ klávesu ENTER.....'+'\n')
Pre skonĉenie programu stlaĉ klávesu ENTER.....

''
>>> |
```

Zapis programu do súboru

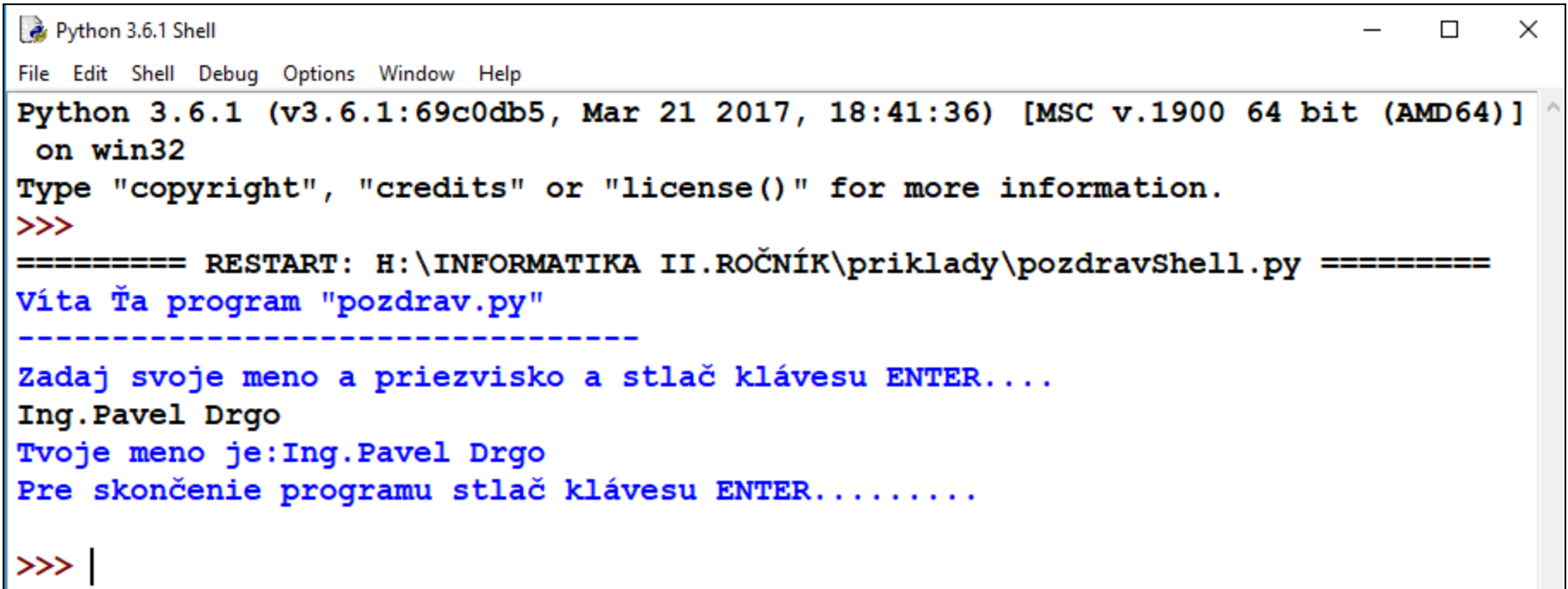
A screenshot of a Python IDE window titled 'pozdravShell.py - H:\INFORMATIKA II.ROČNÍK\priklady\pozdravShell.py (3.6.1)'. The window has a menu bar with 'File', 'Edit', 'Format', 'Run', 'Options', 'Window', and 'Help'. The main area contains Python code with syntax highlighting:

```
print('Víta Ťa program "pozdrav.py"')  
  
print('-----')  
  
meno = input('Zadaj svoje meno a priezvisko a stlač klávesu ENTER....'+'\n')  
  
print('Tvoje meno je:'+ meno)  
  
input('Pre skončenie programu stlač klávesu ENTER.....'+'\n')
```

#-alt+35,alt +x

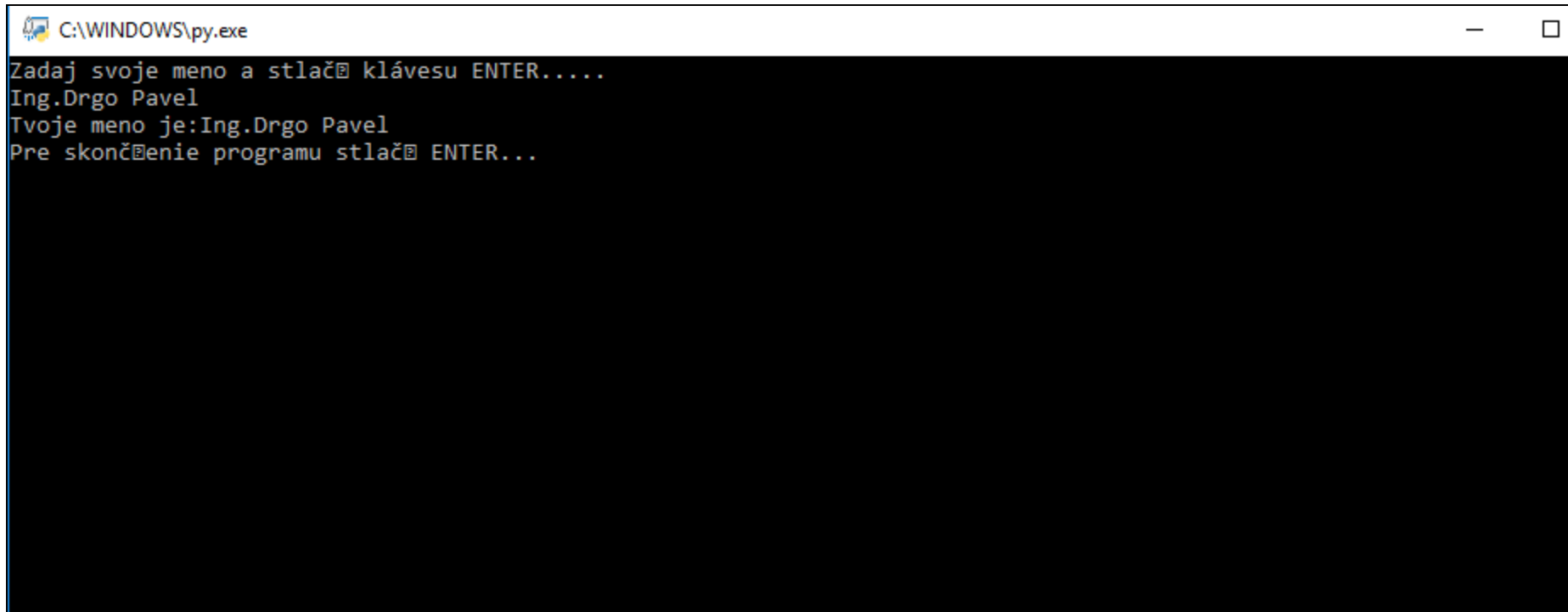
1. `\n` toto spôsobí ,že program po vypísaní hlášky prejde na nový riadok.Tu čaká na vloženie mena
2. Po vložení mena a stlačení ENTER vypíše to čo je v príkaze PRINT

shell



```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: H:\INFORMATIKA II.ROČNÍK\priklady\pozdravShell.py =====
Víta Ťa program "pozdrav.py"
-----
Zadaj svoje meno a priezvisko a stlač klávesu ENTER....
Ing.Pavel Drgo
Tvoje meno je:Ing.Pavel Drgo
Pre skončenie programu stlač klávesu ENTER.....
>>> |
```

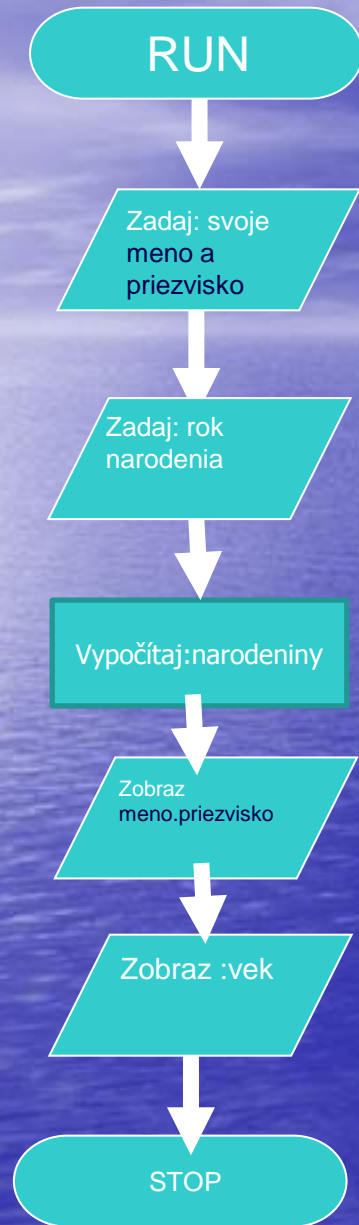
windows



```
C:\WINDOWS\py.exe
Zadaj svoje meno a stlač klávesu ENTER.....
Ing.Drgo Pavel
Tvoje meno je:Ing.Drgo Pavel
Pre skončenie programu stlač ENTER...
```



VÝROČIE



Pr2.vyrocie.py.Zostavte algoritmus na:

- zadanie mena a priezviska
- Zadanie roka narodenia
- Zobrazenie:mena a priezviska
- Zobrazenie:veku

ZADANIE-vyrocie.py

```
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: F:\INFORMATIKA II.ROČNÍK\priklady\vyrocie.py =====
Dobrý deň^, Víta Ťa program výrocie
.....
Zadaj najskôr meno a priezvisko a stlač klávesu ENTER....
Pavel Drgo
Zadaj rok narodenia a stlač klávesu ENTER....
1958
.....
Tvoje meno je:Pavel Drgo
Tento rok by si mal mať:60.narodeniny
Pre skončenie programu stlač ENTER...

>>>
```


Zapis programu

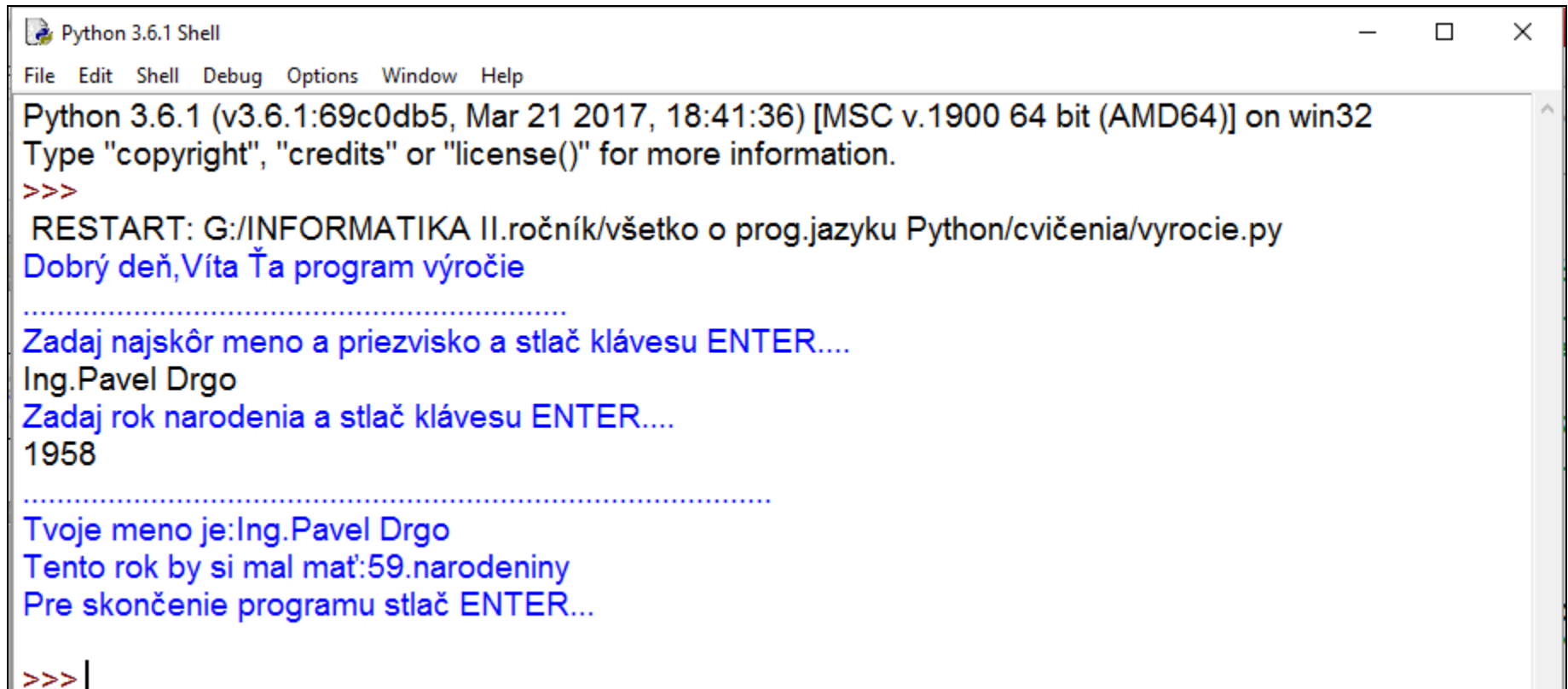
```
1. #Príklad na zadanie a zobrazenie svojho mena a priezviska ,roka narodenia
2. #vyrocie.py
3. print('Dobrý deň,Víta Ťa program výročie')
4. print('.....')
5. meno=input('Zadaj najskôr meno a priezvisko a stlač klávesu ENTER...+'\n')

6. rok=input('Zadaj rok narodenia a stlač klávesu ENTER...+'\n')
7. print('.....')
8. rokCislo=int(rok)
9. narodeniny=2018-rokCislo
10. print('Tvoje meno je:'+meno)
11. print('Tento rok by si mal mať:'+str(narodeniny)+'narodeniny' )
12. input('Pre skončenie programu stlač ENTER...+'\n')
13. #(C)Ing.Drgo Pavel,23.jún 2017,18:25
```

Vyrocie.py

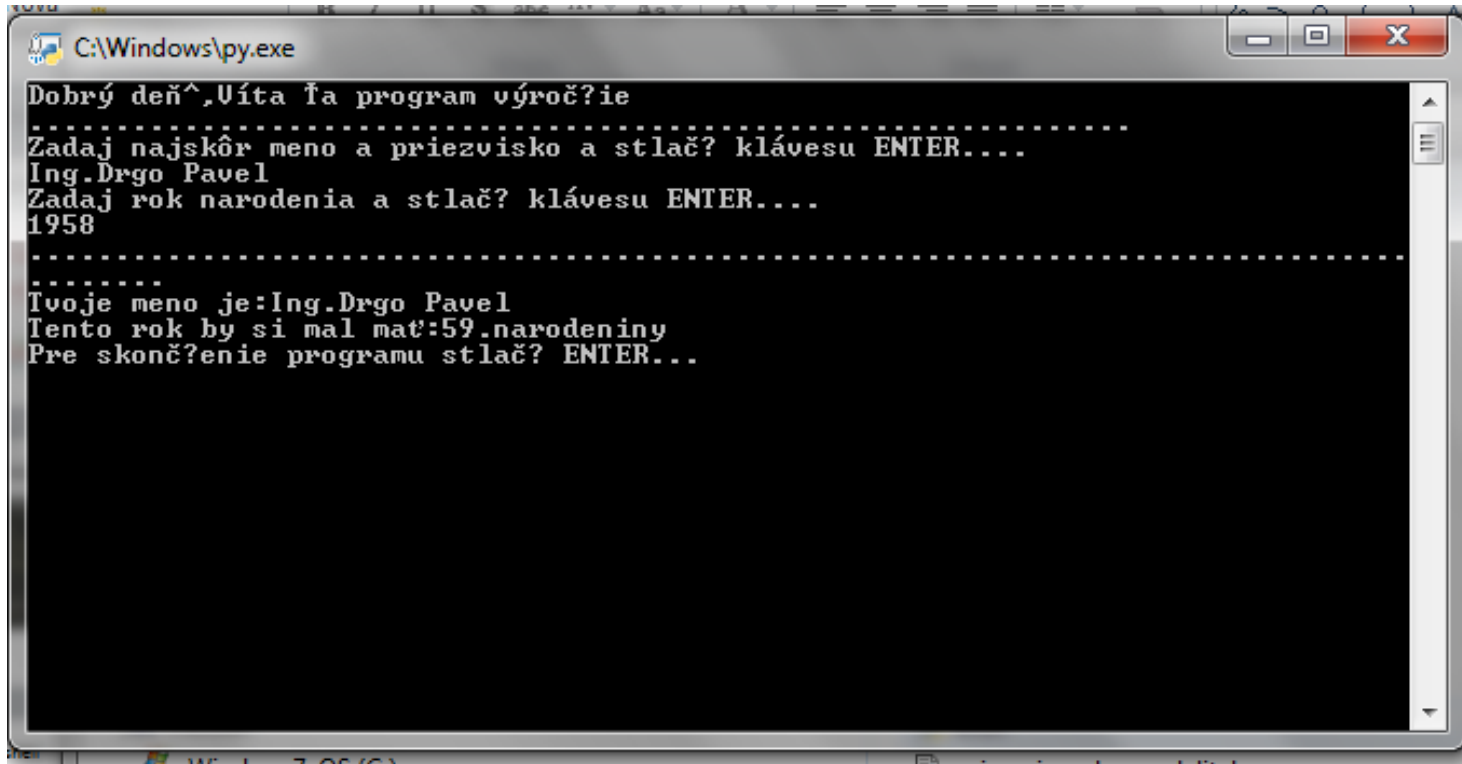
```
vyrocie.py x
1 #Prıklad na zadanie a zobrazenie svojho mena a priezviska ,roka narodenia
2 #vyrocie.py
3 print('Dobrý deň^, Víta Ťa program výročie')
4 print('.....')
5 meno=input('Zadaj najskôr meno a priezvisko a stlač klávesu ENTER....'+'\n')
6
7 rok=input('Zadaj rok narodenia a stlač klávesu ENTER....'+'\n')
8 print('.....')
9 rokCislo=int(rok)
10 narodeniny=2017-rokCislo
11 print('Tvoje meno je:'+meno)
12 print('Tento rok by si mal mať:'+str(narodeniny)+' .narodeniny' )
13 input('Pre skončenie programu stlač ENTER...'+'\n')
14 #(C)Ing.Drgo Pavel,23.jún 2017,18:25
```

shell



```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:/INFORMATIKA II.ročník/všetko o prog.jazyku Python/cvičenia/vyrocie.py
Dobry deň, Víta Ťa program výrocie
.....
Zadaj najskôr meno a priezvisko a stlač klávesu ENTER....
Ing.Pavel Drgo
Zadaj rok narodenia a stlač klávesu ENTER....
1958
.....
Tvoje meno je:Ing.Pavel Drgo
Tento rok by si mal mať:59.narodeniny
Pre skončenie programu stlač ENTER...
>>> |
```

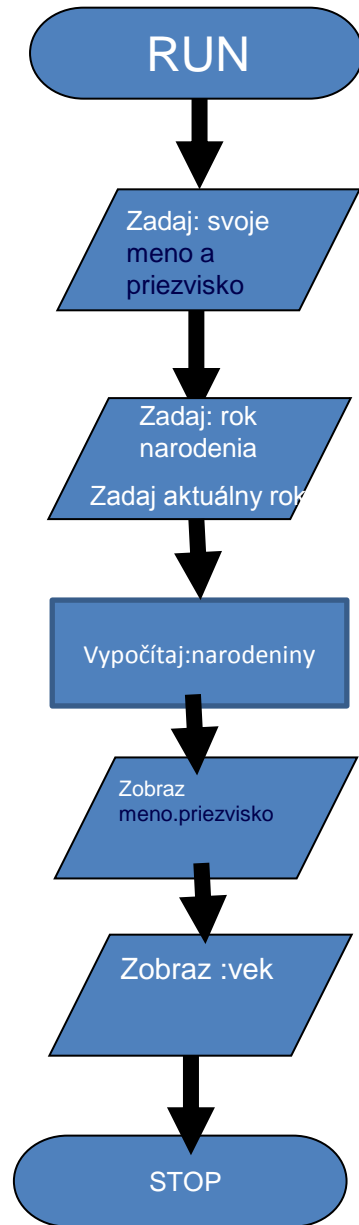
windows



```
C:\Windows\py.exe
Dobrý deň^, Vítá Ťa program úroč?ie
.....
Zadaj najskôr meno a priezvisko a stlač? klávesu ENTER....
Ing.Drgo Pavel
Zadaj rok narodenia a stlač? klávesu ENTER....
1958
.....
Tvoje meno je:Ing.Drgo Pavel
Tento rok by si mal mať:59.narodeniny
Pre skonč?enie programu stlač? ENTER...
```



VÝROČIEUNI



Pr3.vyrocieuni.py.Zostavte algoritmus na:

- **zadanie mena a priezviska**
- **Zadanie roka narodenia**
- **Zadanie aktualneho roka**
- **Zobrazenie:mena a priezviska**
- **Zobrazenie:veku**

Takto má program pracovať:

```
Dobry deň,Víta Ťa program výročie release 2
.....
Zadaj najskôr meno a priezvisko a stlač klávesu ENTER....
Natália Bašková
Zadaj rok narodenia a stlač klávesu ENTER....
2001
Zadaj aktuálny rok a stlač klávesu ENTER....
2018
.....
Tvoje meno je:Natália Bašková
Tento rok by si mal/mala mať:17.narodeniny
Pre skončenie programu stlač ENTER...

>>> |
```

1. *#Príklad na zadanie a zobrazenie svojho mena a priezviska*
2. *#Zadania roka narodenia ,aktuálneho roka,zobrazenie veku*
3. *#vyrocieuni.py*
4. **print('Dobry deň,Víta Ťa program výročíerelease 2')**
5. **print('.....')**
6. **meno=input('Zadaj najskôr meno a priezvisko a stlač klávesu ENTER....'+'\n')**
7. **rokZ=input('Zadaj rok narodenia a stlač klávesu ENTER....'+'\n')**
8. **rokK=input('Zadaj aktuálny rok a stlač klávesu ENTER....'+'\n')**
9. **print('.....')**
10. **rokCisloZ=int(rokZ)**
11. **rokCisloK=int(rokK)**
12. **narodeniny= rokCisloK - rokCisloZ**
13. **print('Tvoje meno je:'+meno)**
14. **print('Tento rok by si mal/mala mať:'+str(narodeniny)+' .narodeniny')**
15. **input('Pre skončenie programu stlač ENTER...'+'\n')**
16. *#(C)Ing.Drgo Pavel,27.marec 2018,16:31,utorok,"SAGAN"*

vyrocieuni.py

```
1 #Prıklad na zadanie a zobrazenie svojho mena a priezviska
2 #Zadania roka narodenia ,aktuálneho roka,zobrazenie veku
3 #vyrocieuni.py
4 print('Dobrý deň,Víta Ťa program výročie release 2')
5 print('.....')
6 meno=input('Zadaj najskôr meno a priezvisko a stlač klávesu ENTER....'+'\n')
7
8 rokZ=input('Zadaj rok narodenia a stlač klávesu ENTER....'+'\n')
9 rokK=input('Zadaj aktuálny rok a stlač klávesu ENTER....'+'\n')
10
11 print('.....')
12 rokCisloZ=int(rokZ)
13 rokCisloK=int(rokK)
14 narodeniny= rokCisloK - rokCisloZ
15 print('Tvoje meno je:'+meno)
16 print('Tento rok by si mal mať:'+str(narodeniny)+' .narodeniny' )
17 input('Pre skončenie programu stlač ENTER...'+'\n')
18 #(C)Ing.Drgo Pavel,27.marec 2018,16:31,utorok,"SAGAN"
```

C:\WINDOWS\py.exe



Dobrý deň,Víta Ťa program výročie release 2

.....

Zadaj najskôr meno a priezvisko a stlač klávesu ENTER....

Natália Bašková

Zadaj rok narodenia a stlač klávesu ENTER....

2001

Zadaj aktuálny rok a stlač klávesu ENTER....

2018

.....

Tvoje meno je: Natália Bašková

Tento rok by si mal/mala mať:17.narodeniny

Pre skončenie programu stlač ENTER...

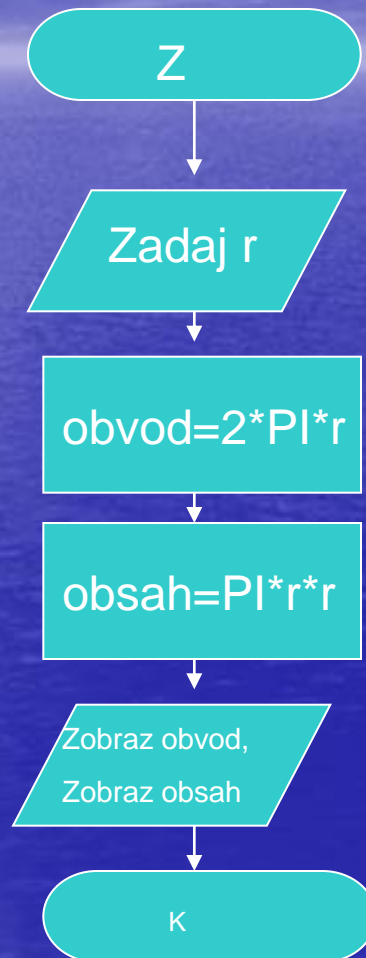
█



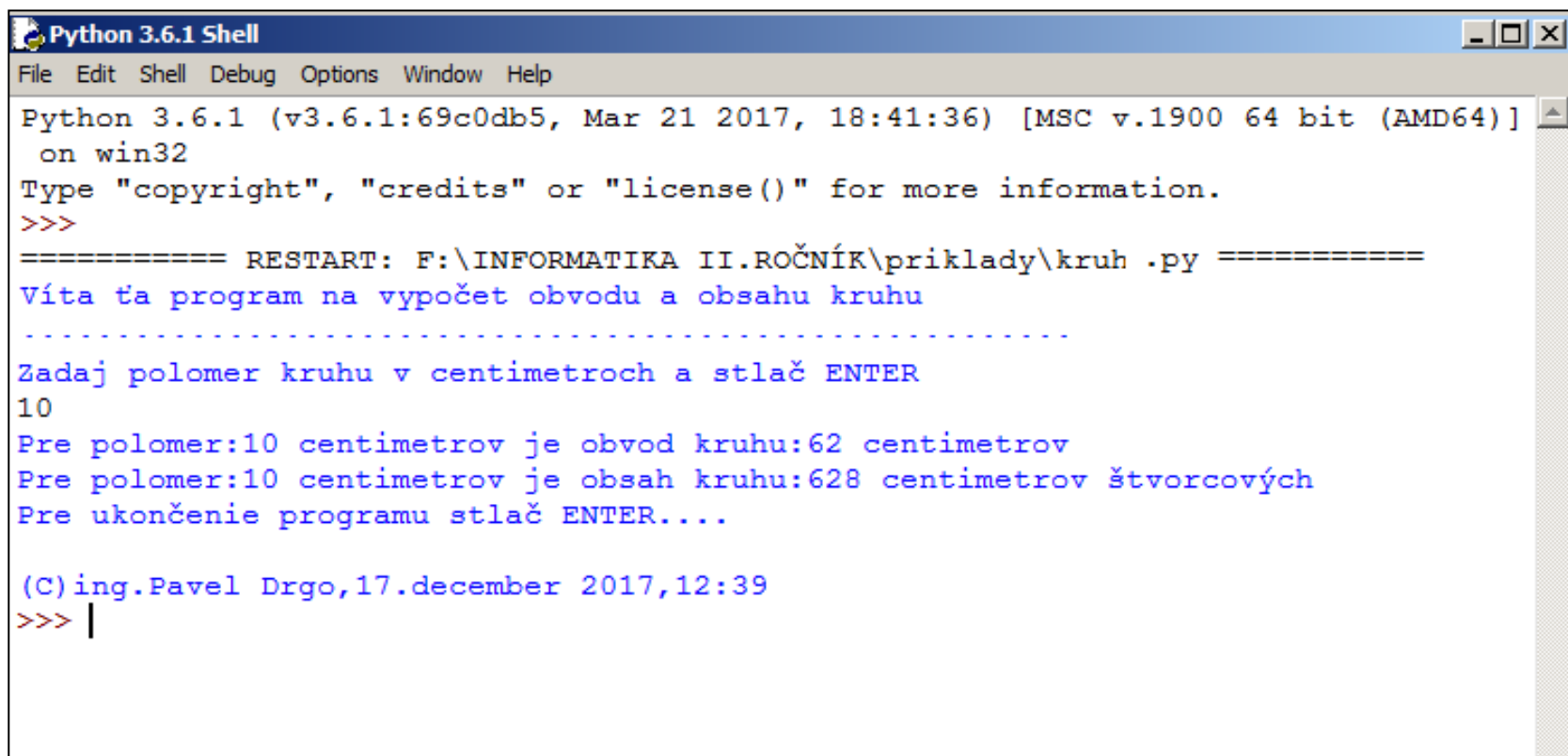
KRUH

Príklady na algoritmickú konštrukciu Sekvencia

Pr.3.Zostavte algoritmus na výpočet obvodu a obsahu kruhu



ZADANIE:kruh.py



```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: F:\INFORMATIKA II.ROČNÍK\priklady\kruh .py =====
Víta ťa program na vypočet obvodu a obsahu kruhu
.....
Zadaj polomer kruhu v centimetroch a stlač ENTER
10
Pre polomer:10 centimetrov je obvod kruhu:62 centimetrov
Pre polomer:10 centimetrov je obsah kruhu:628 centimetrov štvorcových
Pre ukončenie programu stlač ENTER....

(C)ing.Pavel Drgo,17.december 2017,12:39
>>> |
```

```
1. #Program na výpočet obvodu a obsahu kruhu modifikovaný
2. #kruh.py
3. print('Víta t'a program na výpočet obvodu a obsahu kruhu')
4. print('.....')
5. polomer = input('Zadaj polomer kruhu v centimetroch a stlač ENTER'+'\n')
6. polomer = int(polomer)
7. PI=3.14159
8. obvod=2*PI*(polomer)
9. obvod=int(obvod)

10. obsah=2*PI*polomer*polomer

11. obsah=int(obsah)
12. print('Pre polomer:'+str(polomer)+' centimetrov '+'je obvod kruhu:'+
    str(obvod)+' centimetrov ')
13. print('Pre polomer:'+str(polomer)+' centimetrov '+'je obsah
    kruhu:'+str(obsah)+' centimetrov štvorcových')

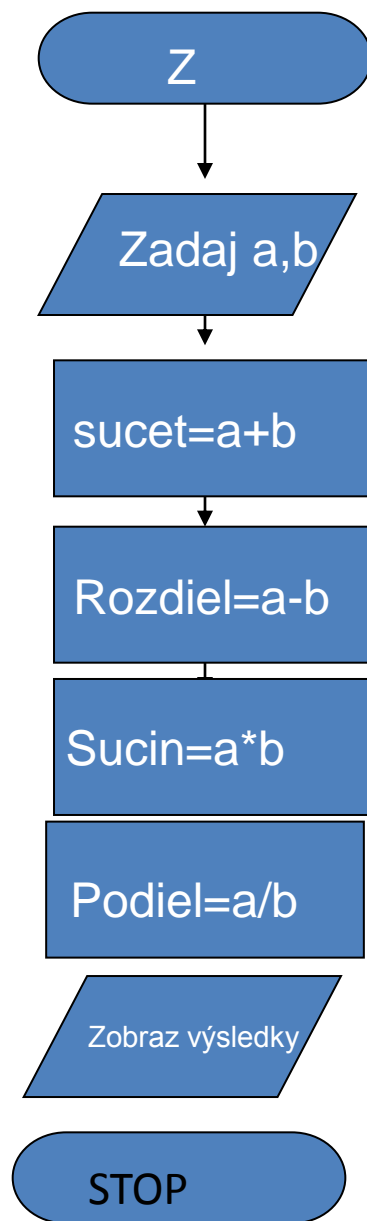
14. input('Pre ukončenie programu stlač ENTER....'+'\n')

15. print('(C)ing.Pavel Drgo,28.február 2018,10:25,streda')
```



JEDNODUCHÁ KALKULAČKA

Pr.4 Naprogramujte jednoduchú kalkulačku



ZADANIE:jednoduchaKalkulacka.py

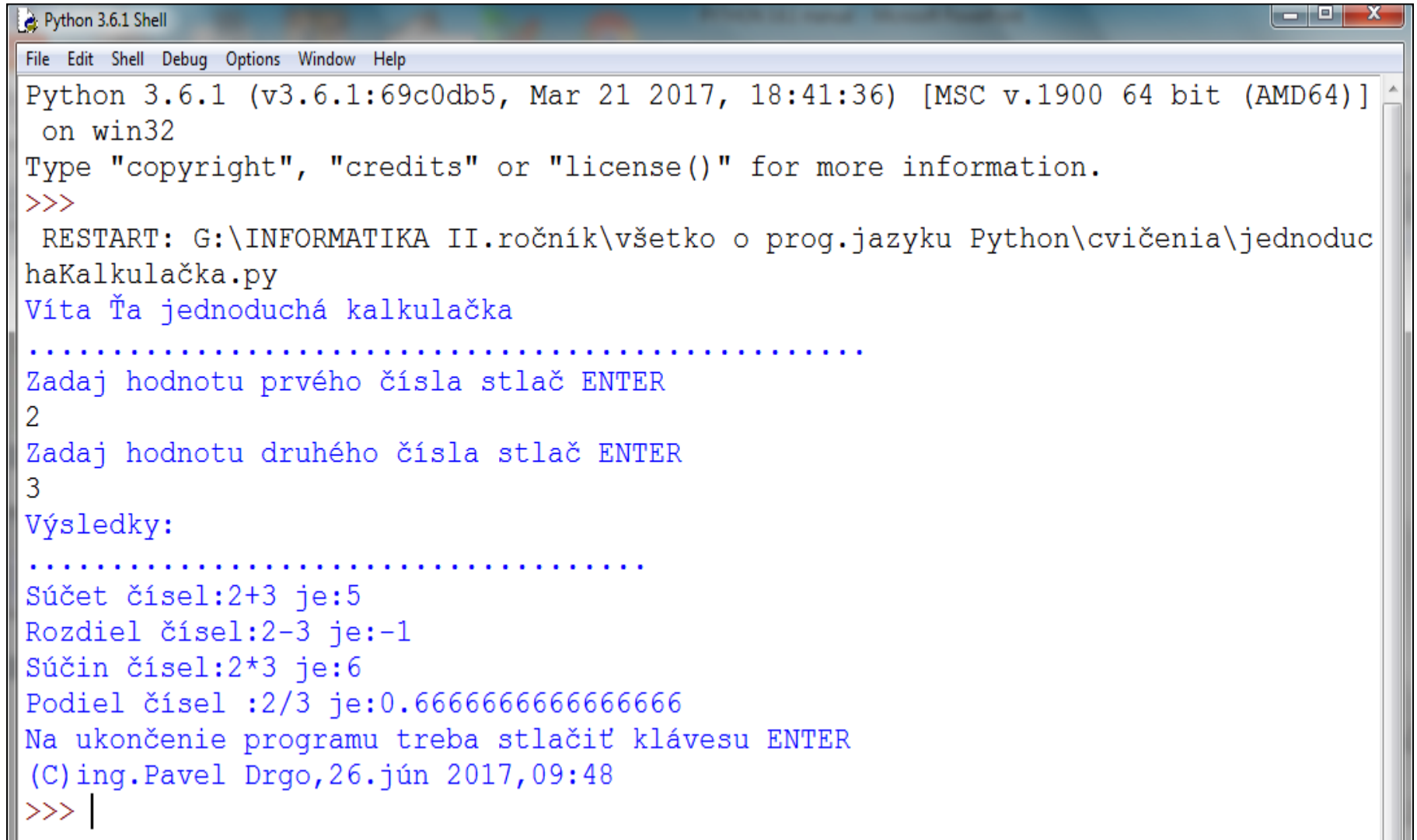
```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: F:\INFORMATIKA II.ROČNÍK\priklady\jednoduchaKalkulacka.py =====
Víta Ťa jednoduchá kalkulačka
.....
Zadaj hodnotu prvého čísla stlač ENTER
2
Zadaj hodnotu druhého čísla stlač ENTER
3
Výsledky:
.....
Súčet čísel:2+3 je:5
Rozdiel čísel:2-3 je:-1
Súčin čísel:2*3 je:6
Podiel čísel :2/3 je:0.6666666666666666
Na ukončenie programu treba stlačiť klávesu ENTER
(C)ing.Pavel Drgo,28.január 2018,12:46
>>> |
```

1. *#Program na základné matematické operácie*
2. *#jednoducháKalkulačka.py*
3. **print**('Víta Ťa jednoduchá kalkulačka')
4. **print**('.....')
5. A=**input**('Zadaj hodnotu prvého čísla stlač ENTER'+'\n')
6. cisloA=**int**(A)
7. B=**input**('Zadaj hodnotu druhého čísla stlač ENTER'+'\n')
8. cisloB=**int**(B)
9. sucet=cisloA+cisloB
10. rozdiel=cisloA-cisloB
11. sucin=cisloA*cisloB
12. podiel=cisloA/cisloB
13. **print**('Výsledky:')
14. **print**('.....')
15. **print**('Súčet čísel:'+str(cisloA)+'+'+str(cisloB)+' je:'+str(sucet))
16. **print**('Rozdiel čísel:'+str(cisloA)+'-'+str(cisloB)+' je:'+str(rozdiel))
17. **print**('Súčin čísel:'+str(cisloA)+'*'+str(cisloB)+' je:'+str(sucin))
18. **print**('Podiel čísel :'+str(cisloA)+'/'+str(cisloB)+' je:'+str(podiel))
19. **input**('Na ukončenie programu treba stlačiť klávesu ENTER')
20. **print**('©ing.Pavel Drgo,28.január 2018,12:46,

jednoduchaKalkulacka.py

```
File Edit Selection View Tools Project Preferences Help
jednoduchaKalkulacka.py
1 #Program na základné matematické operácie
2 #jednoduchaKalkulačka.py
3 print('Víta Ťa jednoduchá kalkulačka')
4 print('.....')
5 A=input('Zadaj hodnotu prvého čísla stlač ENTER'+'\n')
6 cisloA=int(A)
7 B=input('Zadaj hodnotu druhého čísla stlač ENTER'+'\n')
8 cisloB=int(B)
9 sucet=cisloA+cisloB
10 rozdiel=cisloA-cisloB
11 sucin=cisloA*cisloB
12 podiel=cisloA/cisloB
13 print('Výsledky:')
14 print('.....')
15 print('Súčet čísel:'+str(cisloA)+'+'+str(cisloB)+' je:'+str(sucet))
16 print('Rozdiel čísel:'+str(cisloA)+'-'+str(cisloB)+' je:'+str(rozdiel))
17 print('Súčin čísel:'+str(cisloA)+'*'+str(cisloB)+' je:'+str(sucin))
18 print('Podiel čísel :'+str(cisloA)+'/'+str(cisloB)+' je:'+str(podiel))
19 input('Na ukončenie programu treba stlačiť klávesu ENTER')
20 print('(C)ing.Pavel Drgo,28.január 2018,12:46')
21
```

shell



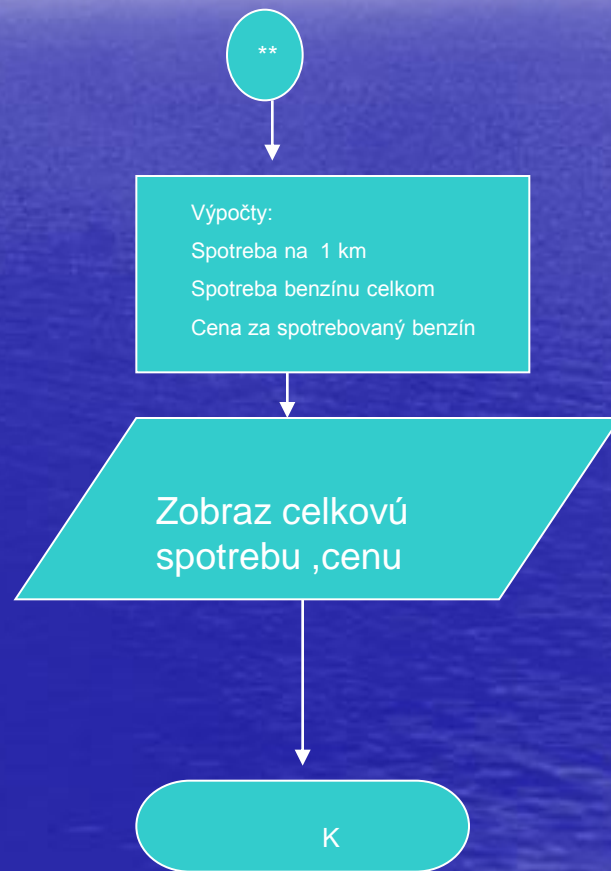
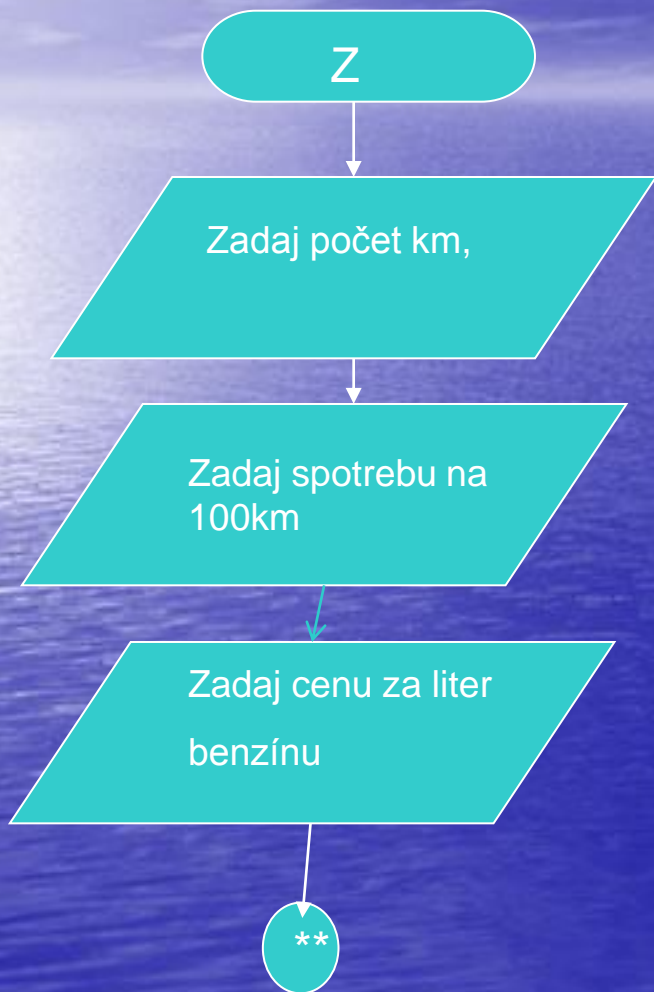
```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:\INFORMATIKA II.ročník\všetko o prog.jazyku Python\cvičenia\jednoducaKalkulačka.py
Víta Ľa jednoduchá kalkulačka
.....
Zadaj hodnotu prvého čísla stlač ENTER
2
Zadaj hodnotu druhého čísla stlač ENTER
3
Výsledky:
.....
Súčet čísel:2+3 je:5
Rozdiel čísel:2-3 je:-1
Súčin čísel:2*3 je:6
Podiel čísel :2/3 je:0.6666666666666666
Na ukončenie programu treba stlačiť klávesu ENTER
(C)ing.Pavel Drgo,26.jún 2017,09:48
>>> |
```



AUTOMOBIL

Príklady na algoritmickú konštrukciu Sekvencia

Pr.5.Zostavte algoritmus na výpočet spotreby benzínu pre automobil



```
1. #Program na výpočet spotreby automobilu základná verzia- nepoužívať
2. #automobil.py
3. print('Víta Ťa auto kalkulačka')
4. print('.....')
5. pk=input('Zadaj počet kilometrov stlač ENTER'+'\n')
6. pocetKilometrov=int(pk)
7. sns= input('Zadaj spotrebu auta na 100 kilometrov stlač ENTER'+'\n')
8. spotrebaNasto=float(sns)
9. czl=input('Zadaj jednotkovú cenu paliva stlač ENTER'+'\n')
10.cenaZaliter=float(czl)
11.spotrebaNajedenkm= spotrebaNasto/100
12.spotrebaCelkom=pocetKilometrov*spotrebaNajedenkm
13.cenaCelkom=cenaZaliter*spotrebaCelkom
14.print('Výsledky:')
15.print('.....')
16.print('Na:'+str(pocetKilometrov)+' kilometrov '+'sa
    minie:'+str(spotrebaCelkom)+'litrov benzínu/nafty')
17.print('Za:'+str(spotrebaCelkom)+' litrov '+'zaplatíš:'+str(cenaCelkom)+'euro')
18.print('(C)ing.Pavel Drgo,30.január 2018,18:54,utorok')
19. input('Na ukončenie programu treba stlačiť klávesu ENTER,...'+'\n')
```

- *#Program na výpočet spotreby automobilu.....upravené výstupy cez int*
- *#automobil.py*
- **print('Víta Ťa auto kalkulačka')**
- **print('.....')**
- **pk=input('Zadaj počet kilometrov stlač ENTER'+'\n')**
- **pocetKilometrov=int(pk)**
- **sns= input('Zadaj spotrebu auta na 100 kilometrov stlač ENTER'+'\n')**
- **spotrebaNasto=float(sns)**
- **czl=input('Zadaj jednotkovú cenu paliva stlač ENTER'+'\n')**
- **cenaZaliter=float(czl)**
- **spotrebaNajedenkm= spotrebaNasto/100**
- **spotrebaCelkom=pocetKilometrov*spotrebaNajedenkm**
- **cenaCelkom=cenaZaliter*spotrebaCelkom**
- **print('Výsledky:')**
- **print('.....')**
- **print('Na:'+str(pocetKilometrov)+' kilometrov '+ 'sa minie:'+str(int(spotrebaCelkom))+ 'litrov benzínu/nafty')**
- **print('Za:'+str(int(spotrebaCelkom))+ ' litrov '+ 'zaplatíš:'+str(int(cenaCelkom))+ 'euro')**
- **print('(C)ing.Pavel Drgo,30.január 2018,18:54,utorok')**
- **input('Na ukončenie programu treba stlačiť klávesu ENTER,...'+'\n')**

automobil.py

```
1 #Program na na výpočet spotreby automobilu
2 #automobil.py
3 print('Víta Ťa auto kalkulačka')
4 print('.....')
5 pk=input('Zadaj počet kilometrov stlač ENTER'+'\n')
6 pocetKilometrov=int(pk)
7 sns= input('Zadaj spotrebu auta na 100 kilometrov stlač ENTER'+'\n')
8 spotrebaNasto=float(sns)
9 czl=input('Zadaj jednotkovú cenu paliva stlač ENTER'+'\n')
10 cenaZaliter=float(czl)
11 spotrebaNajedenkm= spotrebaNasto/100
12 spotrebaCelkom=pocetKilometrov*spotrebaNajedenkm
13 cenaCelkom=cenaZaliter*spotrebaCelkom
14 print('Výsledky:')
15 print('.....')
16 print('Na:'+str(pocetKilometrov)+' kilometrov '+sa minie:'+str(spotrebaCelkom)+'litrov benz:
17 print('Za:'+str(spotrebaCelkom)+' litrov '+zaplatíš:'+str(cenaCelkom)+'euro')
18 print('(C)ing.Pavel Drgo,30.január 2018,18:54,utorok')
19
```

shell

```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:\INFORMATIKA II.ročník\všetko o prog.jazyku Python\cvičenia\automobi
l.py
Víta Ľa auto kalkulačka
.....
Zadaj počet kilometrov stlač ENTER
300
Zadaj spotrebu auta na 100 kilometrov stlač ENTER
7.5
Zadaj jednotkovú cenu paliva stlač ENTER
1.23
Výsledky:
.....
Na:300 kilometrov sa minie:22.5litrov benzínu/nafty
Za:22.5 litrov zaplatíš:27.675euro
(C)ing.Pavel Drgo,26.jún 2017,10:29
>>> |
```

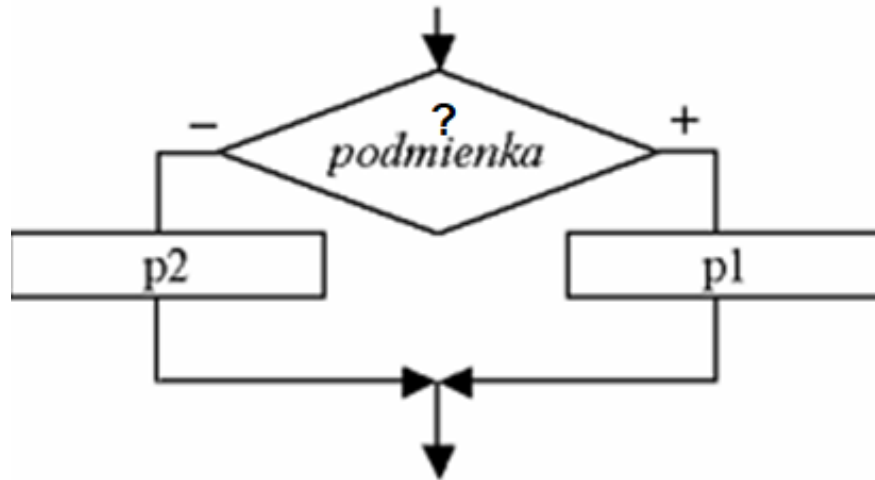
PRESKÚŠANIE

1. Uvedte na čo slúži príkaz **print**
2. Ďalej uvedte na čo slúži príkaz **input**
3. Uvedte príkaz pre premenu číselnej premennej na znakovú
4. Ďalej napíšte príkaz na premenu znakovovej premennej na číselnú
5. Napíšte riadok programu na výpis Vášho mena na obrazovku
6. Napíšte program na zadanie a sčítanie dvoch čísel

Hodnotenie

body	známka
5	1
4	2
3	3
2-1	4
0	5

Vetvenie v programe

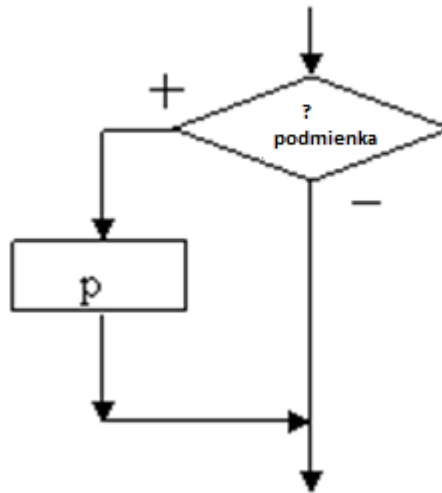


- Vetvenie - sa používa, ak vykonanie príkazu (alebo skupiny príkazov) je podmienené splnením určitej podmienky. (príkaz p1)
- Nesplnenie danej podmienky môže viesť k vykonaniu inej skupiny príkazov.(p2)

Vetvenie

Neúplné vetvenie

Vetvenie môže byť aj neúplné, t.j. ak je podmienka splnená, vykoná sa príkaz(p), ak nie je, nevykoná sa nič.



NEÚPLNÉ VETVENIE

Syntax pre podmienený príkaz (príkaz vetvenia) **if**. Jeho zápis vyzerá takto:

if podmienka: # ak je podmienka splnená, (true), vykonaj prikaz1,2

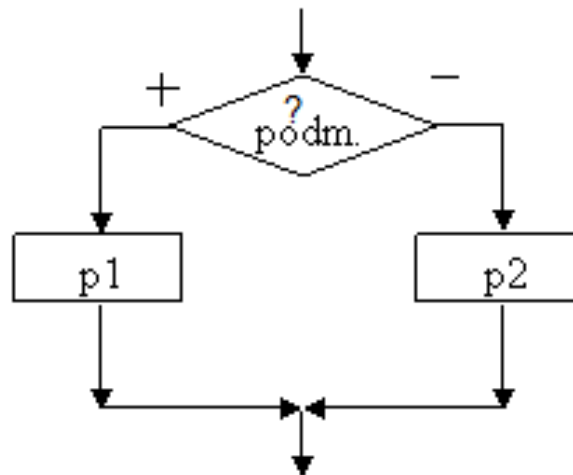
→ prikaz1

→ prikaz2

.....ak podmienka nie je splnená preskočia sa príkazy
(prikaz1, príkaz2) a program pokračuje na tomto riadku

Úplné vetvenie

- Ak je podmienka splnená, program pokračuje vetvou "+",
- ak nie je splnená, pokračuje vetvou "-".



ÚPLNÉ VETVENIE

Syntax pre podmienený príkaz (príkaz vetvenia) **if**. Jeho zápis vyzerá takto:

if podmienka: # ak podmienka platí, vykonaj prikaz1 a príkaz 2

→prikaz1

→prikaz2

...

else: # ak podmienka neplatí, vykonaj prikazy 3a 4

→prikaz3

→prikaz4

...

VIACNÁSObNÉ VETVENIE

V pythone existuje konštrukcia, ktorá uľahčuje vnorenú sériu if-ov:

if podmienka_1: # ak podmienka_1. platí, vykonaj 1. skupinu

↳ príkazov1

↳ príkaz2

...

elif podmienka_2: # ak podmienka_1. neplatí, ale platí podmienka_2, .

↳ Prikaz 3.

↳ príkaz4

...

elif podmienka_3: # ak podmienka_1. a podmienka_2 .neplatia, ale platí podmienka_3

↳ príkaz 5.

↳ príkaz 6

...

else: # ak žiadna z podmienok neplatí, ...

↳ Prikaz 7

...

Vetvenie

V Pythone môžeme zapisovať podmienky podobne, ako je to bežné v matematike:

1. $\text{body} < 90$ je menšie ako
2. $\text{body} \leq 50$ je menšie alebo rovné
3. $\text{body} == 50$ rovná sa
4. $\text{body} != 77$ nerovná sa
5. $\text{body} > 100$ je väčšie ako
6. $\text{body} \geq 90$ je väčšie alebo rovné
7. $40 < \text{body} \leq 50$ je väčšie ako ... a zároveň menšie alebo rovné ...
8. $a < b < c$ a je menšie ako b a zároveň je b menšie ako c

Logické podmienky

4.1 logické operácie

Pozrime sa podrobnejšie na logické operácie `and`, `or` a `not`. Tieto operácie samozrejme fungujú pre logické hodnoty `True` a `False`.

Logický súčin a súčasne:

A	B	A and B
False	False	False
True	False	False
False	True	False
True	True	True

Logický súčet alebo:

A	B	A or B
False	False	False
True	False	True
False	True	True
True	True	True

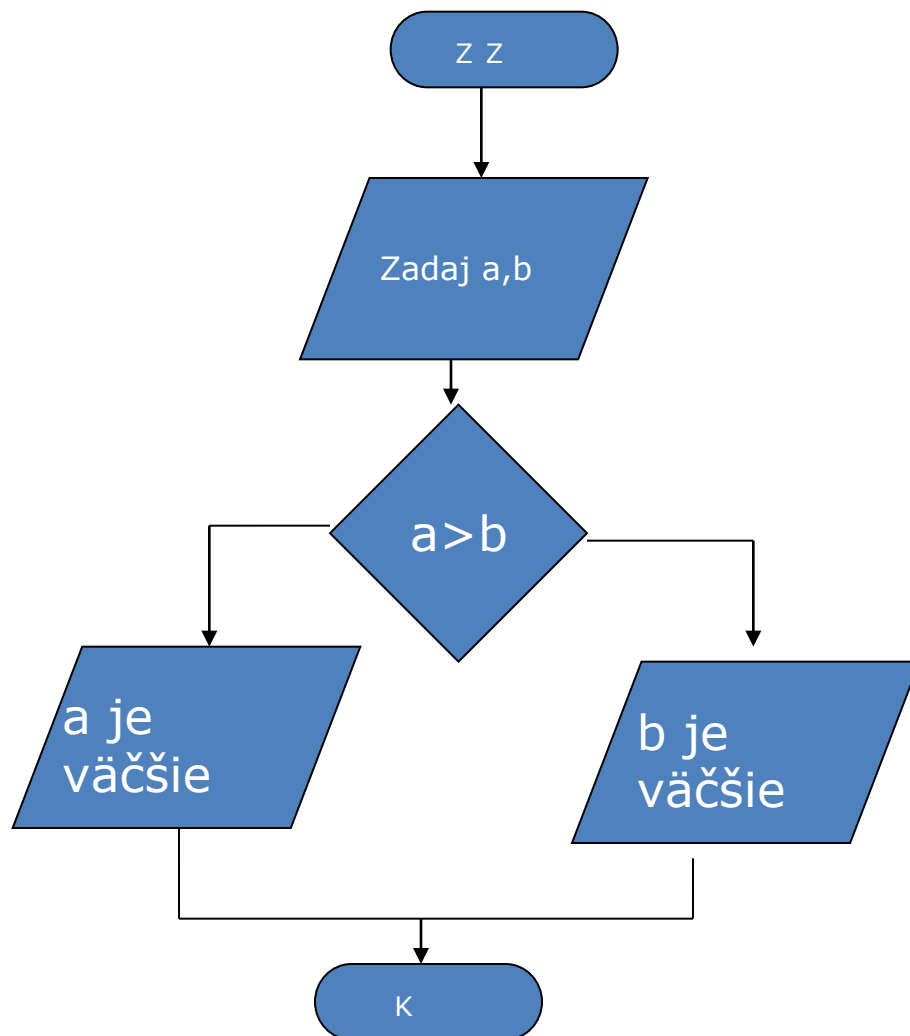
Negácia:

A	not A
False	True
True	False



POROVNAJ DVE ČÍSLA

Príklad 6, porovnávanie dvoch rôznych čísel



```
#Program na porovnanie dvoch čísel  
#porovnajDvecisla  
#Kôli jednoduchosti programu neriešime príklad rovnosti čísel
```

```
print('Víta Ťa komparačný program')
```

```
print('.....')
```

```
A = input('Zadaj prvé s porovnávaných čísel stlač ENTER'+'\n')
```

```
cisloA=int(A)
```

```
B = input('Zadaj druhé s porovnávaných čísel stlač ENTER'+'\n')
```

```
cisloB=int(B)
```

```
if (cisloA > cisloB):
```

```
    print('Prvé číslo:'+ A +' je väčšie ako druhé číslo: '+ B)
```

```
else:
```

```
    print('Druhé číslo:'+ B +' je väčšie ako prvé číslo: '+ A)
```

```
input('Pre ukončenie programu stlač ENTER..'+'\n')
```

```
print('(C)ing.Pavel Drgo,25.február 2018,13:05,nedeľa')
```

```
#Koniec programu
```

porovnajDvecisla.py

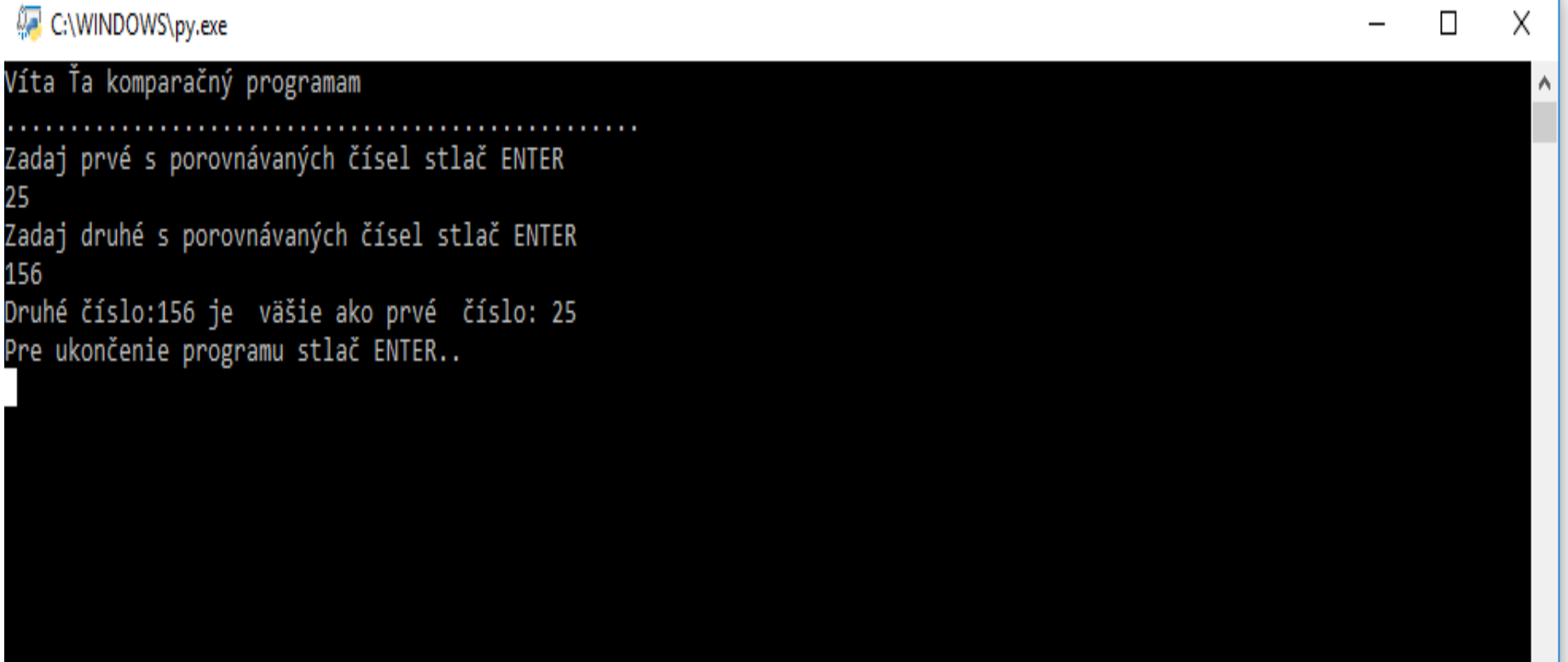
```
1 #Program na porovnanie dvoch čísel
2 #porovnajDvecisla
3 #Kôli jednoduchosti programu neriešime príklad rovnosti čísel
4 print('Víta Ťa komparačný programam')
5
6 print('.....')
7
8 A = input('Zadaj prvé s porovnávaných čísel stlač ENTER'+'\n')
9
10 cisloA=int(A)
11 B = input('Zadaj druhé s porovnávaných čísel stlač ENTER'+'\n')
12 cisloB=int(B)
13
14 if (cisloA > cisloB):
15     print('Prvé číslo:'+ A +' je väčšie ako druhé číslo: '+ B)
16
17 else:
18     print('Druhé číslo:'+ B +' je väčšie ako prvé číslo: '+ A)
19
20
21 input('Pre ukončenie programu stlač ENTER..'+'\n')
22 print('(C)ing.Pavel Drgo,25.február 2018,13:05,nedeľa')
23 #Koniec programu
24
```


shell

```
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
  on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: G:\INFORMATIKA II.ROČNÍK\priklady\porovnajDvecisla.py =====
Víta Ťa komparačný programam
.....
Zadaj prvé s porovnávaných čísel stlač ENTER
12
Zadaj druhé s porovnávaných čísel stlač ENTER
58
Druhé číslo:58 je väčšie ako prvé číslo: 12
Pre ukončenie programu stlač ENTER..

(C)ing.Pavel Drgo,25.február 2018,13:05,nedeľa
>>>
```

windows

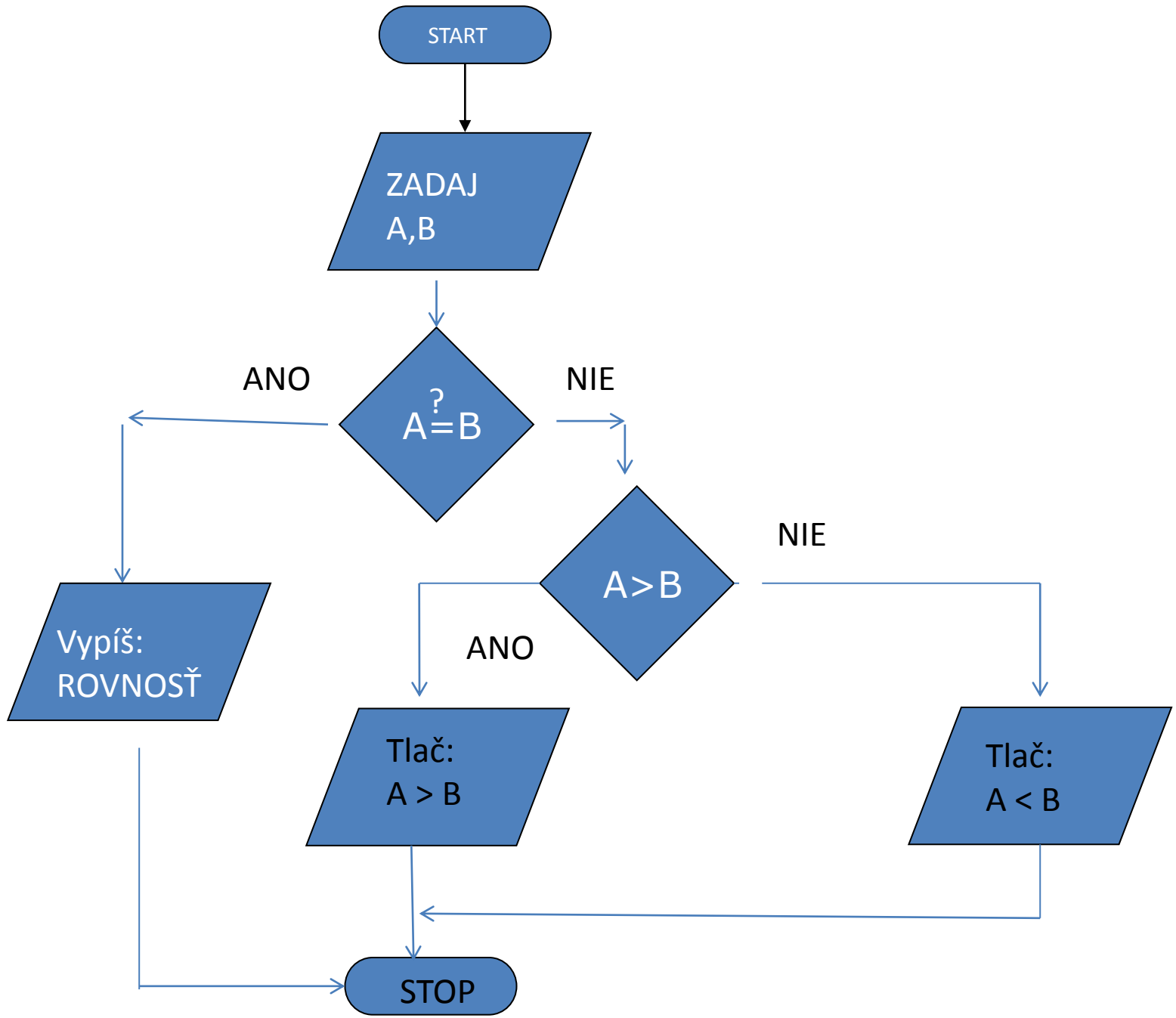


A screenshot of a Windows command prompt window. The title bar shows the file path "C:\WINDOWS\py.exe" and standard window controls (minimize, maximize, close). The command prompt displays the following text:

```
Víta Ťa komparačný programam  
.....  
Zadaj prvé s porovnávaných čísel stlač ENTER  
25  
Zadaj druhé s porovnávaných čísel stlač ENTER  
156  
Druhé číslo:156 je väčšie ako prvé číslo: 25  
Pre ukončenie programu stlač ENTER..
```



KOMPARÁTOR



```
#Program na porovnanie dvoch čísel súbor: porovnajDvecislaRovnost.py
#Tu je vyriešená aj rovnosť zadaných čísel
print('Víta Ťa komparačný programam')
print('.....')
A = input('Zadaj prvé s porovnávaných čísel stlač ENTER'+'\n')
cisloA=int(A)
B = input('Zadaj druhé s porovnávaných čísel stlač ENTER'+'\n')
cisloB=int(B)

if (cisloA > cisloB) and (cisloA != cisloB):

    print('Prvé číslo:'+ A +' je väčšie ako druhé číslo: '+ B)

elif (cisloA < cisloB) and (cisloA != cisloB):
    print('Druhé číslo:'+ B +' je väčšie ako prvé číslo: '+ A)
else :
print('Prvé číslo:'+ A +' je rovnaké ako druhé číslo: '+ B)

input('Pre ukončenie programu stlač ENTER..'+'\n')
print('(C)ing.Pavel Drgo,25.február 2018,13:05,nedeľa')
#Koniec programu
```

```
1 #Program na porovnanie dvoch čísel súbor: porovnajDvecislaRovnost.py
2 #porovnajDvecisla
3 #Tu je vyriešená aj rovnosť zadaných čísel
4 print('Víta Ťa komparačný programam')
5
6 print('.....')
7
8 A = input('Zadaj prvé s porovnávaných čísel stlač ENTER'+'\n')
9
10 cisloA=int(A)
11 B = input('Zadaj druhé s porovnávaných čísel stlač ENTER'+'\n')
12 cisloB=int(B)
13
14 if (cisloA > cisloB) and (cisloA != cisloB):
15     print('Prvé číslo:'+ A +' je väčšie ako druhé číslo: '+ B)
16
17 elif (cisloA < cisloB) and (cisloA != cisloB):
18     print('Druhé číslo:'+ B +' je väčšie ako prvé číslo: '+ A)
19 else :
20     print('Prvé číslo:'+ A +' je rovnaké ako druhé číslo: '+ B)
21
22 input('Pre ukončenie programu stlač ENTER..'+'\n')
23 print('(C)ing.Pavel Drgo,25.február 2018,13:05,nedeľa')
24 #Koniec programu
25
26
```

```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
=== RESTART: G:\INFORMATIKA II.ROČNÍK\priklady\porovnajDvecislaRovnost.py ===
Víta Ťa komparačný programam
.....
Zadaj prvé s porovnávaných čísel stlač ENTER
23
Zadaj druhé s porovnávaných čísel stlač ENTER
12
Prvé číslo:23 je väčšie ako druhé číslo: 12
Pre ukončenie programu stlač ENTER..

(C)ing.Pavel Drgo,25.február 2018,13:05,nedeľa
>>>
```

C:\WINDOWS\py.exe

Víta Āa komparaĀnĀ programam

.....

Zadaj prvé s porovnávanĀch Āísel stlaĀ ENTER

458

Zadaj druhé s porovnávanĀch Āísel stlaĀ ENTER

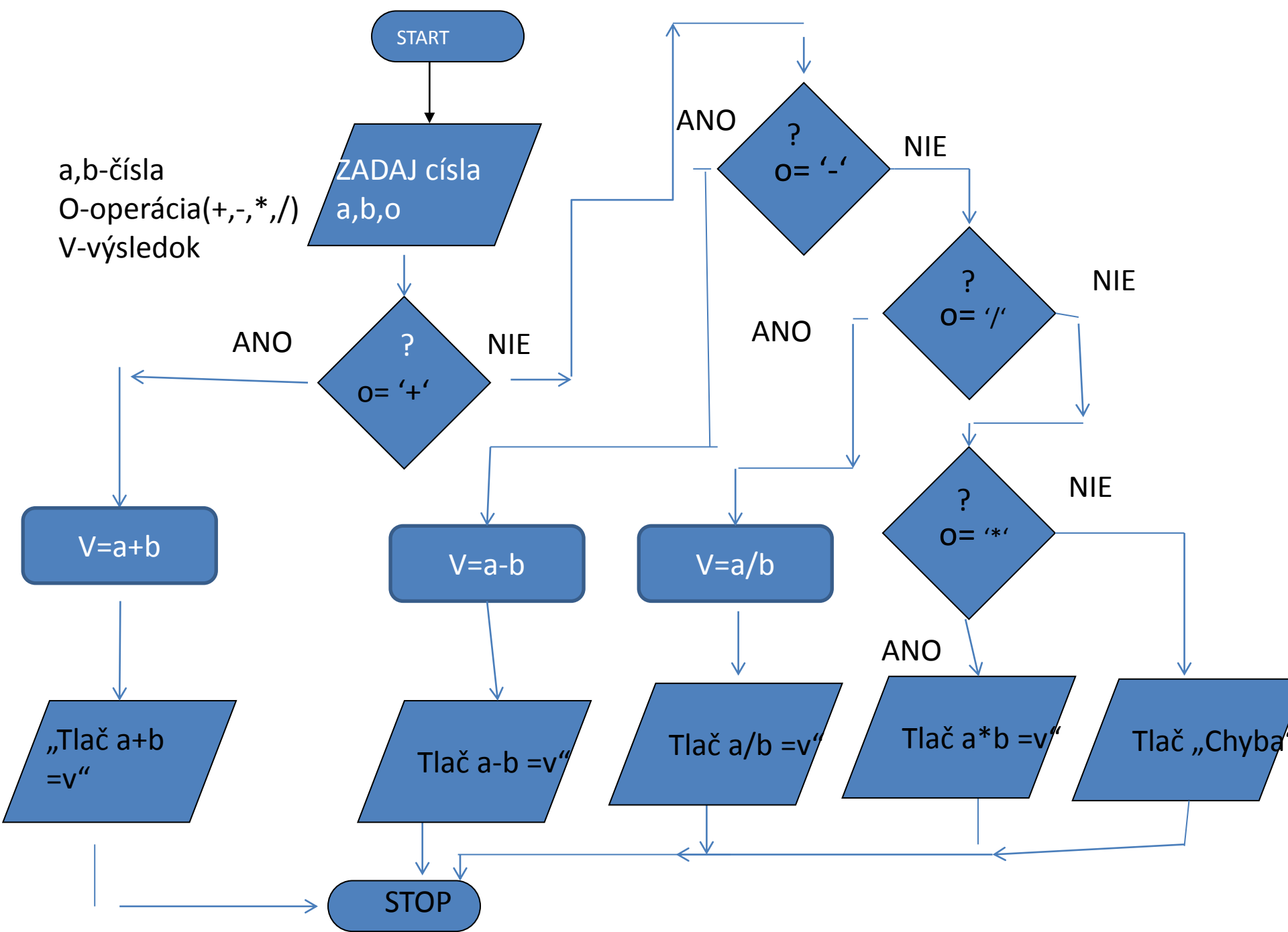
256

Prvé Āíslo:458 je vĀšie ako druhé Āíslo: 256

Pre ukonĀenie programu stlaĀ ENTER..



kalkulačka



Ukážka programu

```
Víta Ťa program kalkulačka
```

```
-----
```

```
Vyber si matematickú operáciu (+,-,*,/)..a stlač ENTER
```

```
*
```

```
Zadaj prvé číslo a stlač ENTER...
```

```
52
```

```
Zadaj druhé číslo a stlač ENTER...
```

```
2
```

```
52*2 je:104
```

```
Pre skončenie programu stlač klávesu ENTER...
```

```
#Príklad na viacnásobné vetvenie kalkulacka2.py
#Treba zadať čísla pre operáciu a značku operácie
#Privítanie
print('Víta Ťa program kalkulačka')
print('-----')
#Zadávanie údajov
operacia=input('Vyber si matematickú operáciu (+,-,*,/)..a stlač ENTER'+'\n')
a = int(input('Zadaj prvé číslo a stlač ENTER...'+'\n'))
b = int(input('Zadaj druhé číslo a stlač ENTER...'+'\n'))
#Vyhodnotenie zvolenej operácie
if(operacia=='+'):
    v=a+b
    print(str(a)+operacia+str(b)+' je:'+str(v))
elif(operacia=='-'):
    v=a-b
    print(str(a)+operacia+str(b)+' je:'+str(v))
elif(operacia=='*'):
    v=a*b
    print(str(a)+operacia+str(b)+' je:'+str(v))
elif(operacia=='/'):
    v=a/b
    print(str(a)+operacia+str(b)+' je:'+str(int(v)))
else:
    print('Vybral si:'+operacia+' čo je chyba!!!'+ 'spusti program znova')
#Ukončenie programu
input(' Pre skončenie programu stlač klávesu ENTER...'+'\n')
#Ing.Drgo Pavel,16.marec 2018,piatok,18:07,nainštaloval som 30Mb/s internet)
```



GENERÁTOR POZDRAVU

#Program na pozdravenie podľa času

#pozdravCas.py

print('Víta Ťa generátor pozdravu')

print('.....')

vstup=input('Zadaj počet hodín v rozsahu 0-23 a stlač ENTER'+'\n')

cas=int(vstup)#Zmena reťazca na číslo

if (cas >= 6)and(cas<=10):

print('Dobré ráno')

elif (cas>=11)and(cas<=18):

print('Dobrý deň')

elif (cas>=19)and(cas<=22):

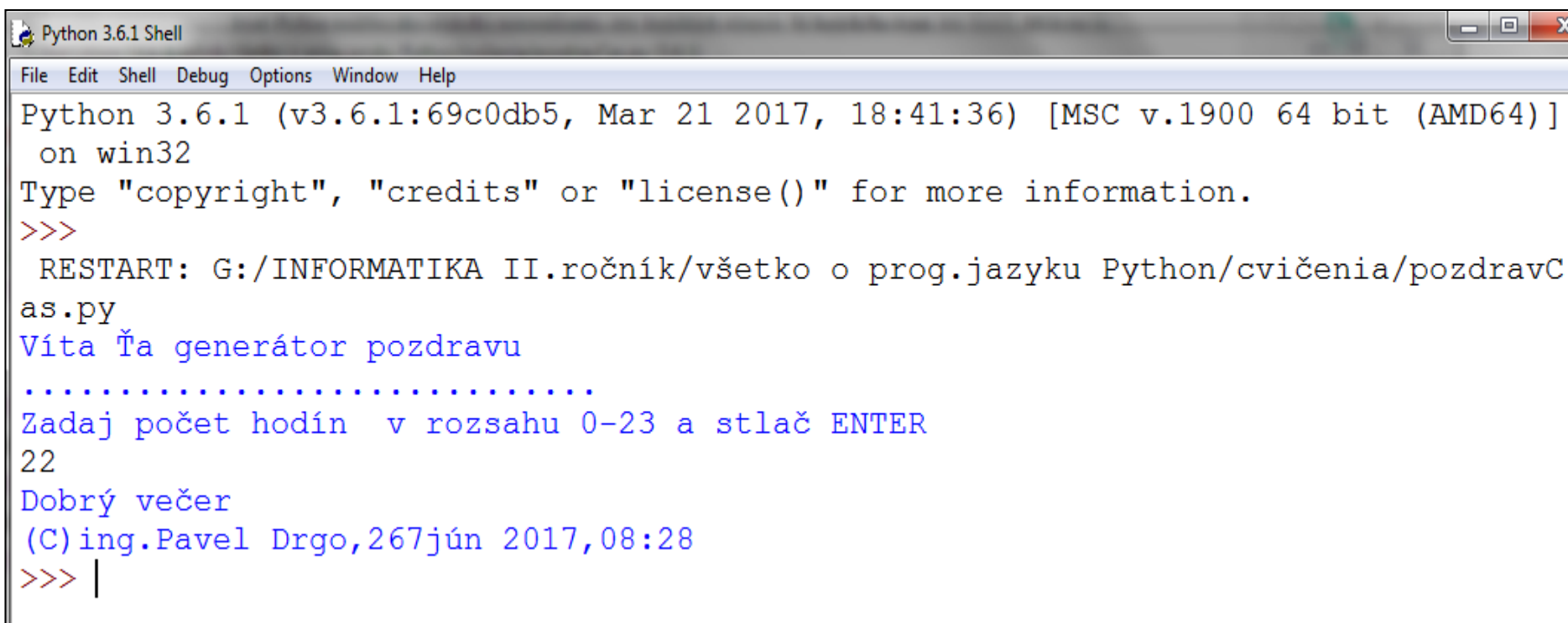
print('Dobrý večer')

else:

print('Dobrá noc')

print('(C)ing.Pavel Drgo,26 jún 2017,08:28')

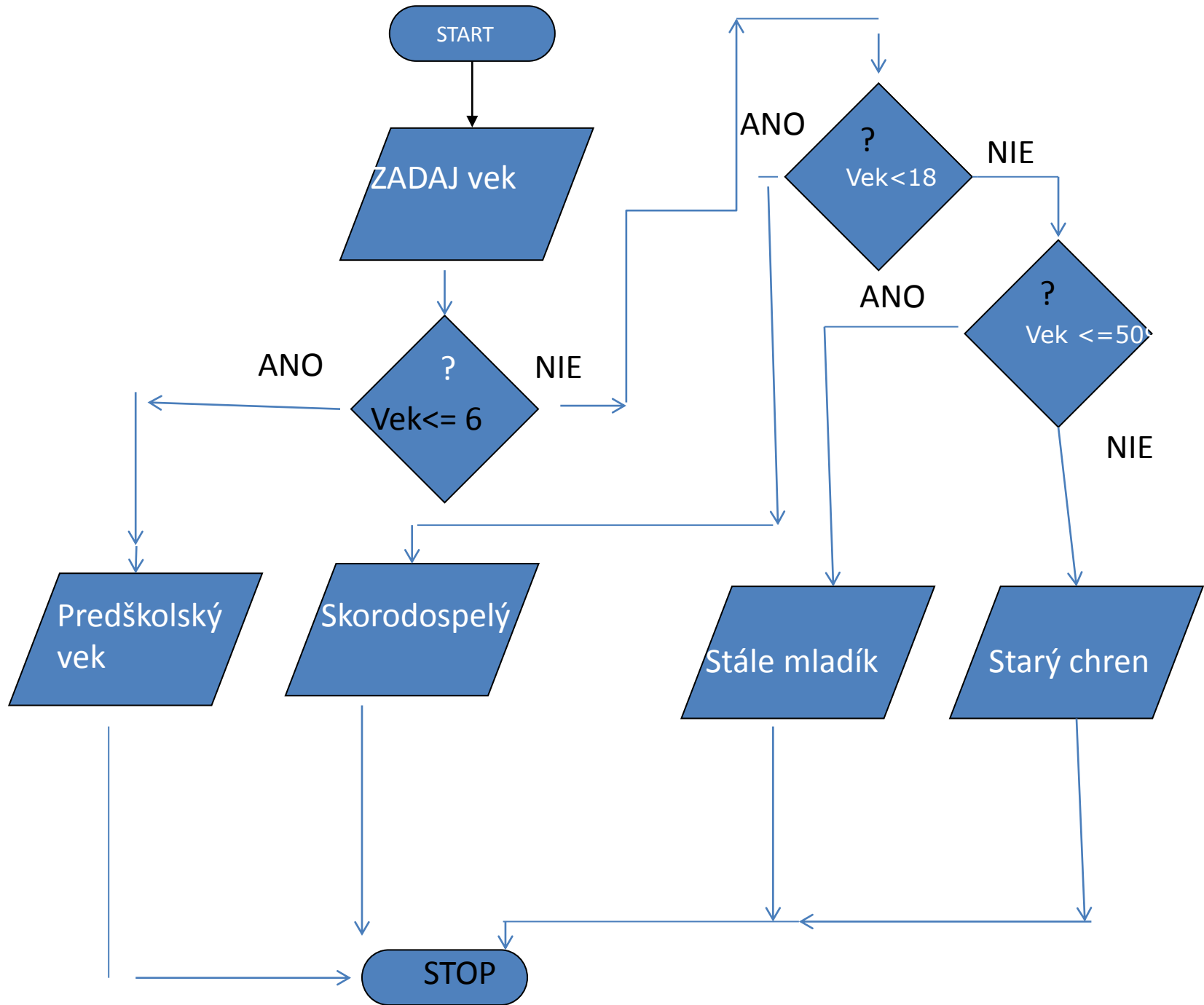
shell



```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:/INFORMATIKA II.ročník/všetko o prog.jazyku Python/cvičenia/pozdravC
as.py
Víta Ľa generátor pozdravu
.....
Zadaj počet hodín v rozsahu 0-23 a stlač ENTER
22
Dobrý večer
(C)ing.Pavel Drgo,267jún 2017,08:28
>>> |
```



VEK



#Program na vyhodnotenie veku zo zadaného roku

#tvojVek.py

print('Víta Ťa generátor pozdravu')

print('.....')

vstup=input('Zadaj svoj vek a stlač ENTER'+'\n')

vek=int(vstup)*#Zmena reťazca na číslo*

if (vek<= 6):

print('Si ešte v predškolskom veku')

elif (vek<18):

print('Už si skoro dospelý')

elif (vek<=50):

print('Už nie si mladík')

elif (vek>50):

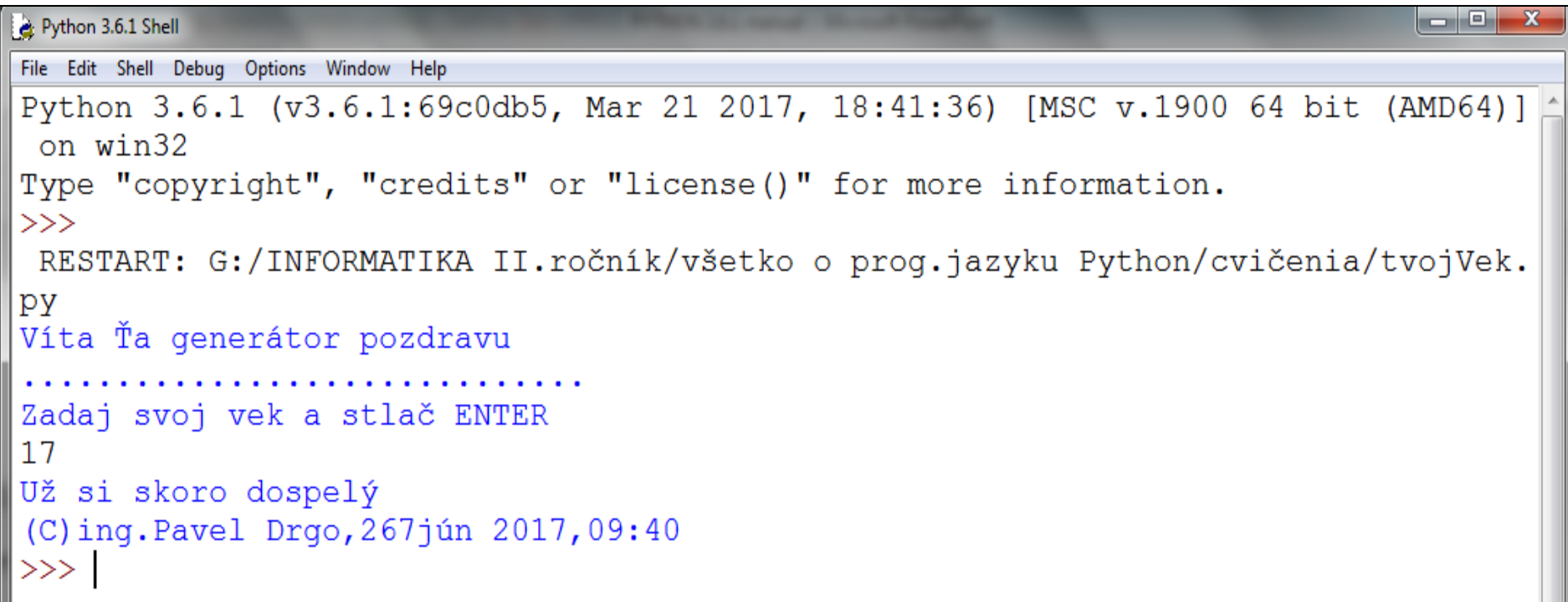
print('Si už starý chren')

else:

print('Spusti znova program')

print('(C)ing.Pavel Drgo,27jún 2017,09:40')

shell

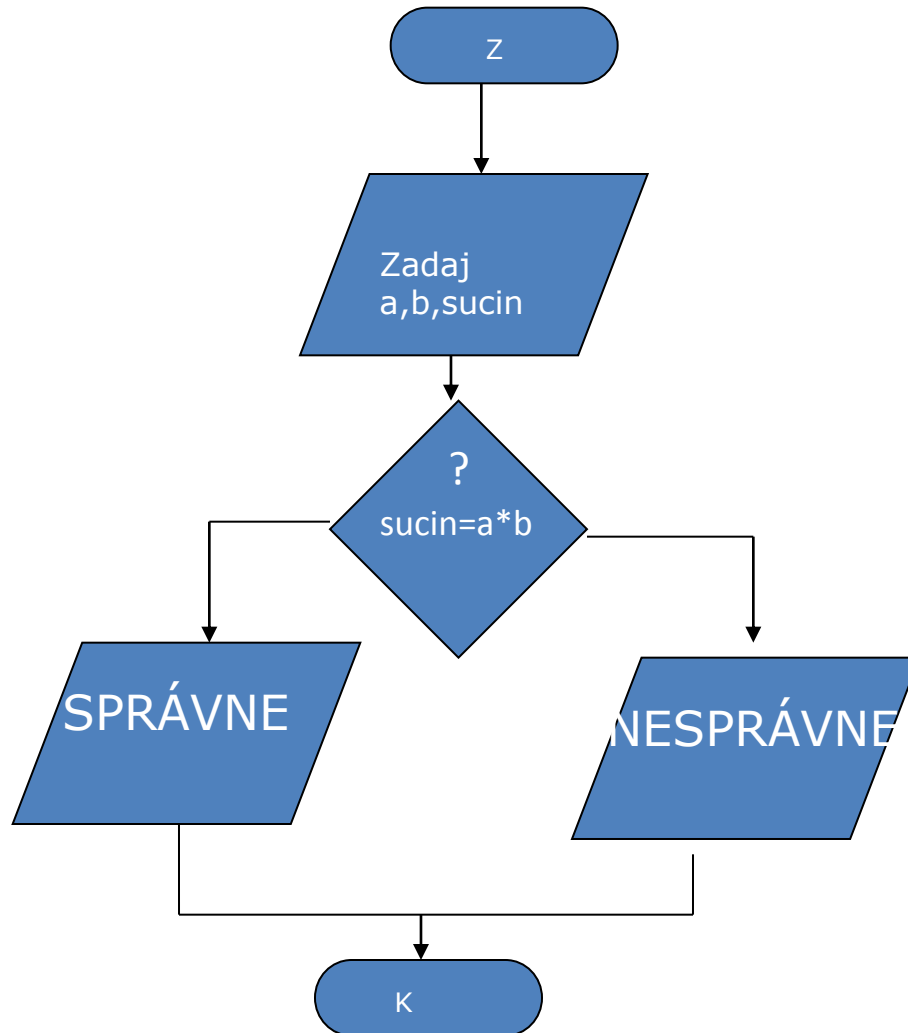


```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:/INFORMATIKA II.ročník/všetko o prog.jazyku Python/cvičenia/tvojVek.
PY
Víta Ľa generátor pozdravu
.....
Zadaj svoj vek a stlač ENTER
17
Už si skoro dospelý
(C)ing.Pavel Drgo,267jún 2017,09:40
>>> |
```



NASOBILKA

Test násobilky



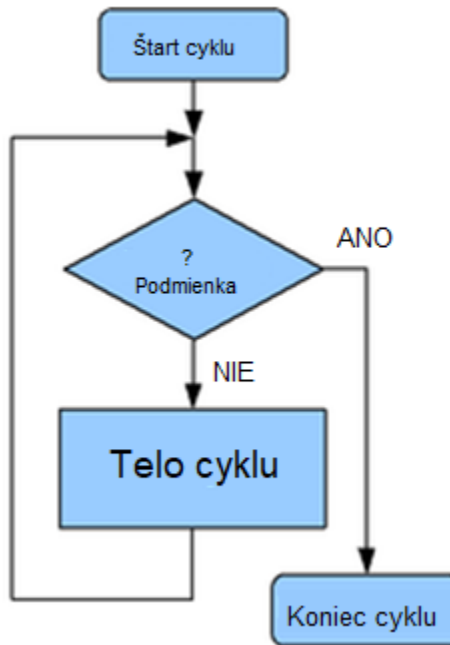
```
#Program na skúšanie násobilky
#nasobilka.py
print('Víta Ťa učiteľ násobilky')
print('.....')
vstup=input('Zadaj prvé číslo násobilky a stlač ENTER'+'\n')
cisloA=int(vstup)#Zmena reťazca na číslo
vstup=input('Zadaj druhé číslo násobilky a stlač ENTER'+'\n')
cisloB=int(vstup)
vstup=input('Zadaj výsledok ,ktorý si myslíš ,že je správny a stlač
ENTER'+'\n')
cisloSucin=int(vstup)
cisloSpravne=cisloA*cisloB
if (cisloSucin==cisloSpravne):
    print('Výborne: '+str(cisloA)+' * '+str(cisloB)+' je skutočne:
'+str(cisloSpravne))

else:
    print('Bohužiaľ nesprávne: '+str(cisloA)+' * '+str(cisloB)+' nie je :
'+str(cisloSucin))
input('Pre ukončenie programu stlač klávesu ENTER')
print('(C)ing.Pavel Drgo,27jún 2017,10:06')
```

shell

```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:/INFORMATIKA II.ročník/všetko o prog.jazyku Python/cvičenia/nasobilk
a.py
Víta Ťa učiteľ násobilky
.....
Zadaj prvé číslo násobilky a stlač ENTER
2
Zadaj druhé číslo násobilky a stlač ENTER
3
Zadaj výsledok ,ktorý si myslíš ,že je správny a stlač ENTER
5
Bohužial nesprávne: 2 * 3 skutočnenie je : 5
Pre ukončenie programu stlač klávesu ENTER
(C)ing.Pavel Drgo,267jún 2017,10:06
>>>
```

Realizácia algoritmickkej konštrukcie cyklus



- Cyklus sa skladá z tela cyklu a podmieneného skoku ,
- Cyklus ukončuje pri splnení podmienky.

Cykly pre dopredu neznámy počet opakovaní

Cyklus **while** je najjednoduchší cyklus v Pythone, jeho syntax je nasledujúca:

```
while podmienka:  
    blok_príkazov  
else:  
    blok_príkazov
```

Blok **while** pokračuje vo vykonávaní príkazov tak dlho, kým platí podmienka.

Ak podmienka neplatí, vykoná sa nepovinný blok **else** .

Blok **else** sa ale nevykoná v prípade, že sa z cyklu vyskočí napr. Príkazom **break** alebo **return**

Do bloku **while** môžeme pridať aj príkazy **break** a **continue**. **Break** z cyklu vyskočí, **continue** v cykle skočí na začiatok bloku príkazov.



DOTIERAVKA1

Príklad použitia cyklu while

```
===== RESTART: G:\INFORMATIKA II.ROČNÍK\priklady\while1.py =====  
Dotieravá aplikácia  
.....  
Pre ukončenie cyklu napíš slovo Python alebo python  
55  
Pre ukončenie cyklu napíš slovo Python alebo python  
adam  
Pre ukončenie cyklu napíš slovo Python alebo python  
no čo je?  
Pre ukončenie cyklu napíš slovo Python alebo python  
python  
Slovo je zadané!  
Program ukončíte stlačením ľubovolnej klávesy...  
  
>>>
```

#Program na demonštrovanie použitia cyklu WHILE

#program while1.py

print("Dotieravá aplikácia")

print('.....')

pokracovat = True

while pokracovat:

slovo = input("Pre ukončenie cyklu napíš slovo Python alebo python "+'\n')

if (slovo == "Python" or slovo == "python"):

print("Slovo je zadané!")

pokracovat = False

else:

pass # Pass

#Kľúčové slovo Pythone pass sa používa v podmienkach (alebo kdekoľvek inde

#pričom znamená, že daná časť kódu nič nerobí.

input("Program ukončíte stlačením ľubovolnej klávesy..."+'\n')

#Ing.Drgo Pavel, 16.marec 2018, 19:27,piatok,protesty proti Robertovi Ficovi



DOTIERAVKA2

```
===== RESTART: G:/INFORMATIKA II.ROČNÍK/priklady/vstup.py =====  
Napíš A: ahoj  
Hovoril som napíš A!!!  
Napíš A: No čo chcel  
Hovoril som napíš A!!!  
Napíš A: a  
Hovoril som napíš A!!!  
Napíš A: Á  
Hovoril som napíš A!!!  
Napíš A: A  
Hurá, je to A!  
>>>
```

vstup.py - G:\INFORMATIKA II.ROČNÍK\prikklady\vstup.py (3.6.1)

File Edit Format Run Options Window Help

```
#Program vstup.py
```

```
vstup = ""
```

```
while vstup != "A":
```

TAB

```
    vstup = input("Napiš A: ")
```

```
        if vstup != "A":
```

TAB

```
            print("Hovoril som napiš A!!!")
```

```
        else :
```

TAB

```
            print("Hurá, je to A!")
```



Kalkulačka s while


```
#Program kalkulackaWhile.py
#Privítanie
print('Víta Ťa program kalkulačka s opakovaním')
print('-----')
pokracovat = True
while pokracovat:
#Zadávanie údajov
    operacia=input('Vyber si matematickú operáciu (+, -, *, /)..a stlač ENTER'+'\n')
    a = int(input('Zadaj prvé číslo a stlač ENTER...'+'\n'))
    b = int(input('Zadaj druhé číslo a stlač ENTER...'+'\n'))
#Vyhodnotenie zvolenej operácie
    if(operacia=='+'):
        v=a+b
        print(str(a)+operacia+str(b)+' je:'+str(v))
    elif(operacia=='-'):
        v=a-b
        print(str(a)+operacia+str(b)+' je:'+str(v))
    elif(operacia=='*'):
        v=a*b
        print(str(a)+operacia+str(b)+' je:'+str(v))
    elif(operacia=='/'):
        v=a/b
        print(str(a)+operacia+str(b)+' je:'+str(int(v)))
    else:
        print('Vybral si:'+operacia+' čo je chyba!!!'+spusti program znova')
#Ukončenie programu
    volba=input(' Chceš opakovať výpočet ?..napíš ano/nie...'+'\n')
    if (volba == "ano" or volba == "áno"):
        pokracovat = True
    else:
        pokracovat= False
input('Pre regulérne ukončenie stlač ľubovoľnú klávesu....'+'\n')
#Ing.Drgo Pavel, 16.marec 2018,piatok, 19:53,radio head awards
```

kalkulackaWhile.py

```
1 #Program kalkulackaWhile.py
2 #Privítanie
3 print('Víta Ťa program kalkulacka s opakovaním')
4 print('-----')
5 pokračovat = True
6 while pokračovat:
7     #Zadávanie údajov
8     operacia=input('Vyber si matematickú operáciu (+,-,*,/)..a stlač ENTER'+'\n')
9     a = int(input('Zadaj prvé číslo a stlač ENTER...'+'\n'))
10    b = int(input('Zadaj druhé číslo a stlač ENTER...'+'\n'))
11    #Vyhodnotenie zvolenej operácie
12    if(operacia=='+'):
13        v=a+b
14        print(str(a)+operacia+str(b)+' je:'+str(v))
15    elif(operacia=='-'):
16        v=a-b
17        print(str(a)+operacia+str(b)+' je:'+str(v))
18    elif(operacia=='*'):
19        v=a*b
20        print(str(a)+operacia+str(b)+' je:'+str(v))
21    elif(operacia=='/'):
22        v=a/b
23    print(str(a)+operacia+str(b)+' je:'+str(int(v)))
24    else:
25        print('Vybral si:'+operacia+' čo je chyba!!!'+spusti program znova')
26    #Ukončenie programu
27    volba=input(' Chceš opakovať výpočet ?..napíš ano/nie...'+'\n')
28    if (volba == "ano" or volba == "áno"):
29        pokračovat = True
30    else:
31        pokračovat= False
32    input('Pre regulérne ukončenie stlač ľubovoľnú klávesu....'+'\n')
33    #Ing.Drigo Pavel,16.marec 2018,piatok,19:53,radio head awards
34
35
```

Cykly pre menší počet opakovaní

SYNTAX:

for počítadlo **in** zoznam hodnôt počítadla:

príkaz 1

 príkaz 2

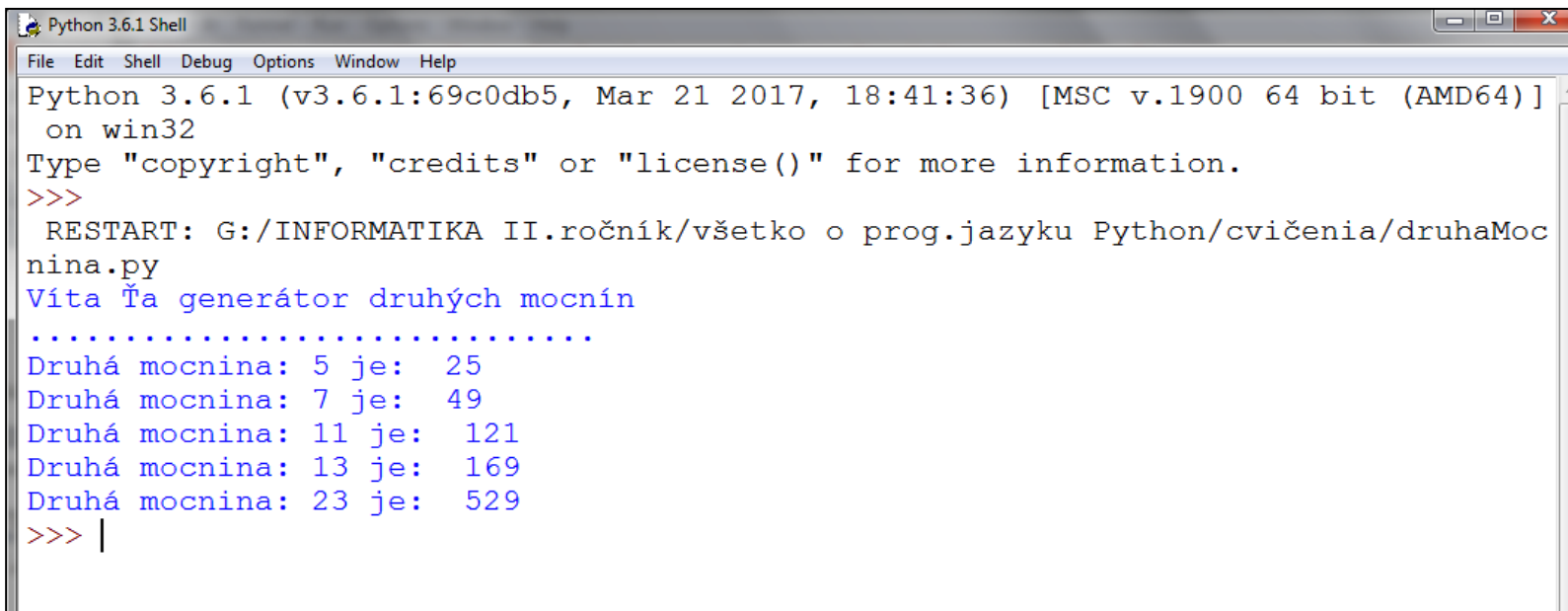
*

*

prikaz n

Příklad9 zostavte program,ktorý t zobrazí druhé mocniny prvočísel

```
#Program na výpočet druhej mocniny zadaných prvočísel  
#druhaMocnina.py  
print('Víta Ěa generátor druhých mocnín')  
print('.....')  
for j in 5, 7, 11, 13, 23:  
    jecko=j**2  
    print('Druhá mocnina:', j, 'je: ', jecko)  
 #(C)ing.Pavel Drgo,27.jún 2017,11:41
```



```
Python 3.6.1 Shell  
File Edit Shell Debug Options Window Help  
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]  
on win32  
Type "copyright", "credits" or "license()" for more information.  
>>>  
RESTART: G:/INFORMATIKA II.ročník/všetko o prog.jazyku Python/cvičenia/druhaMoc  
nina.py  
Víta Ěa generátor druhých mocnín  
.....  
Druhá mocnina: 5 je: 25  
Druhá mocnina: 7 je: 49  
Druhá mocnina: 11 je: 121  
Druhá mocnina: 13 je: 169  
Druhá mocnina: 23 je: 529  
>>> |
```

Generovanie postupnosti čísel

range(start, stop, krok)

Parametre

- • **start** – prvý prvok vygenerovanej postupnosti (ak chýba, predpokladá sa 0)
- • **stop** – hodnota, na ktorej sa už generovanie d'alšej hodnoty postupnosti zastaví – táto hodnota v postupnosti už nebude
- • **krok** – hodnota, o ktorú sa zvýši (resp. zníži pre záporný krok) každý nasledovný prvok postupnosti
ak tento parameter chýba, predpokladá sa 1
- Väčšinou platí, že do parametra **stop** nastavíme o 1 väčšiu hodnotu, ako potrebujeme poslednú hodnotu vygenerovanej postupnosti

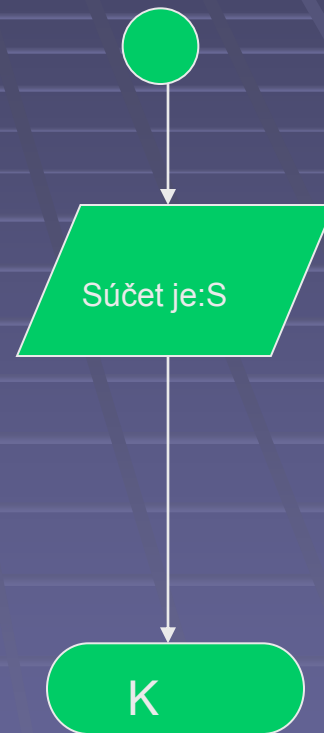
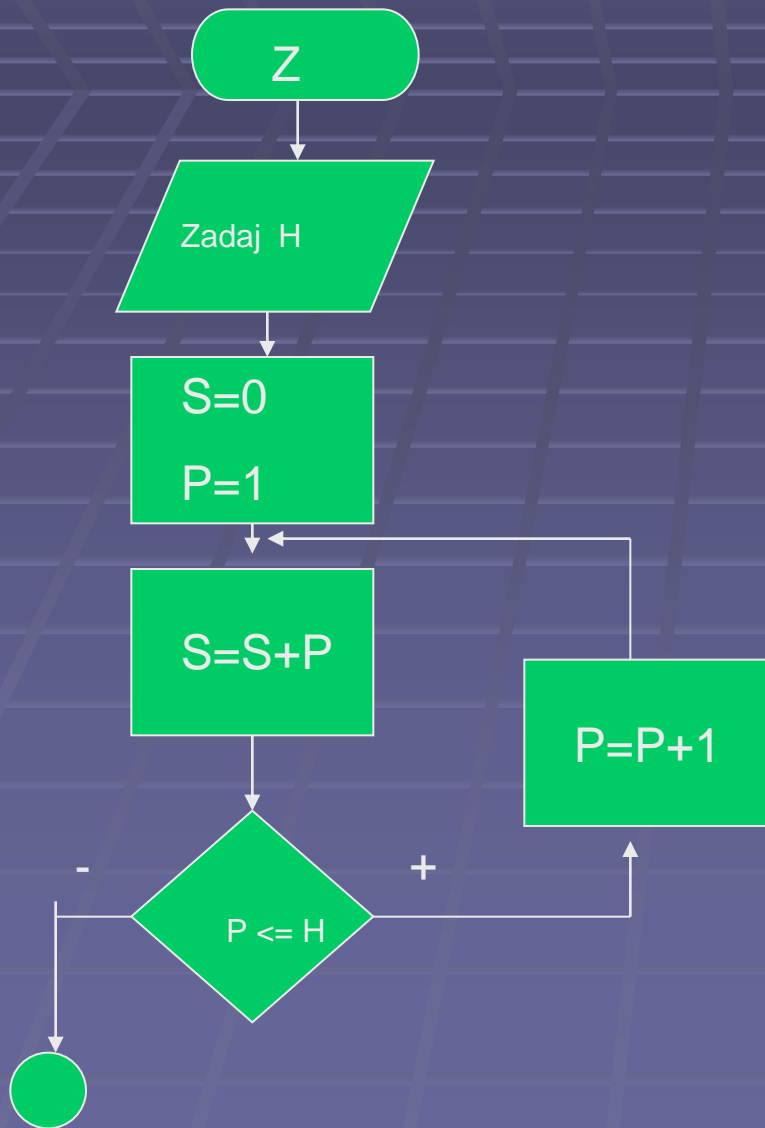
Priklady nastavenia parametrov funkcie `range()`

<code>range (10)</code>	0,1,2,3,4,5,6,7,8,9
<code>range (0, 10)</code>	0,1,2,3,4,5,6,7,8,9
<code>range (0, 10, 1)</code>	0,1,2,3,4,5,6,7,8,9
<code>range (3, 10)</code>	3,4,5,6,7,8,9
<code>range (3, 10, 2)</code>	3,5,7,9
<code>range (10, 100, 10)</code>	10,20,30,40,50,60,70,80,90
<code>range (10, 1, -1)</code>	10,9,8,7,6,5,4,3,2
<code>range (10, 1)</code>	<i>prázdna postupnosť</i>
<code>range (1, 1)</code>	<i>prázdna postupnosť</i>

.....

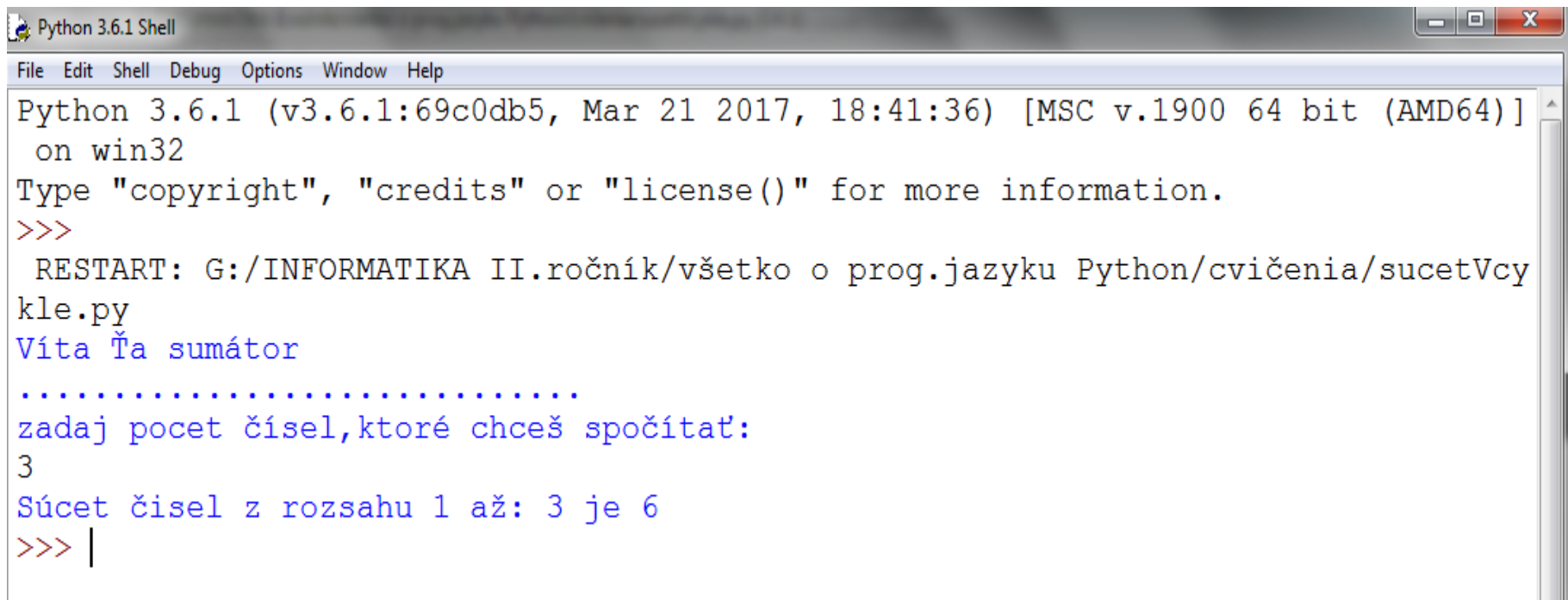
Príklady na algoritmickú konštrukciu FOR

Pr.3. Vytvorte algoritmus na súčet čísel z rozsahu 1 až H




```
#Program na výpočet súčtu čísel z rozsahu  
#SucetVcykle.py  
print('Víta Ěa sumátor')  
print('.....')  
n = int(input('zadaj pocet čísel,ktoré chceš spočítat: '+'\n'))  
sucet = 0  
for i in range(1, n+1):  
    sucet = sucet + i # alebo sucet += i  
print('Súčet čísel z rozsahu 1 až: '+str(n)+' je', str(sucet))  
 #(C)ing.Pavel Drgo,27.jún 2017,12:42
```

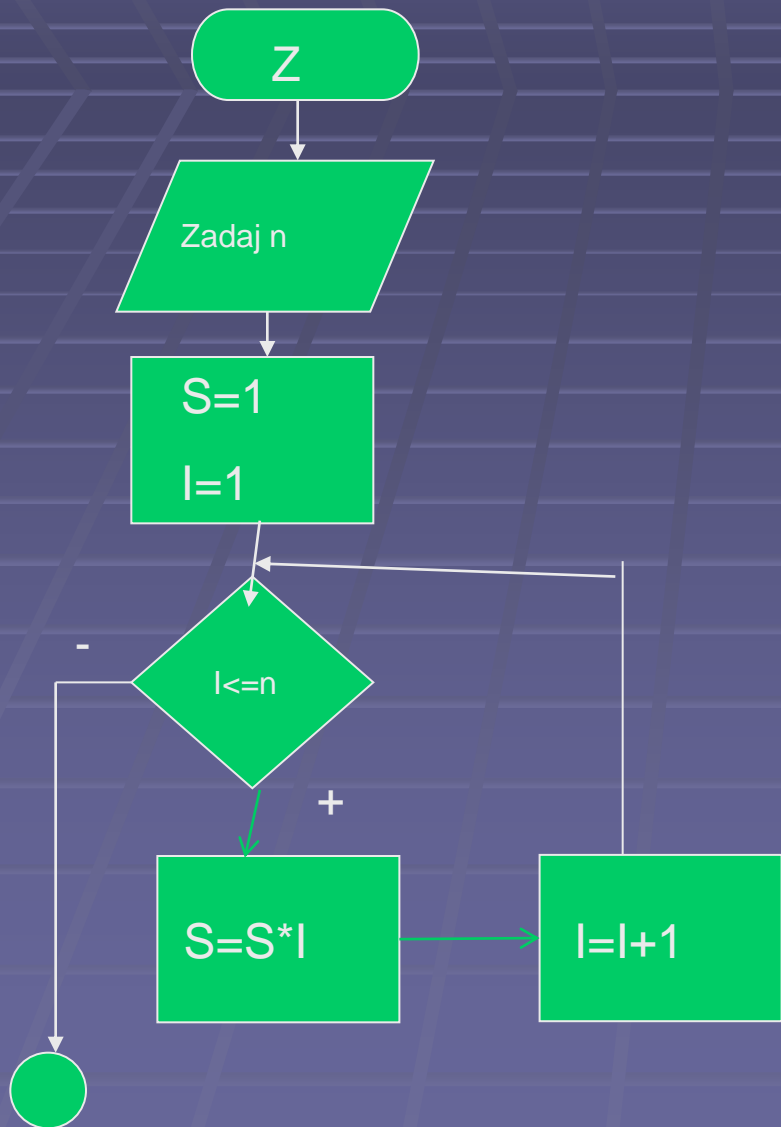
shell



```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
  RESTART: G:/INFORMATIKA II.ročník/všetko o prog.jazyku Python/cvičenia/sucetVcy
kle.py
Víta Ťa sumátor
.....
zadaj pocet čísel, ktoré chceš spočítať:
3
Súčet čísel z rozsahu 1 až: 3 je 6
>>> |
```

Príklady na algoritmickú konštrukciu WHILE

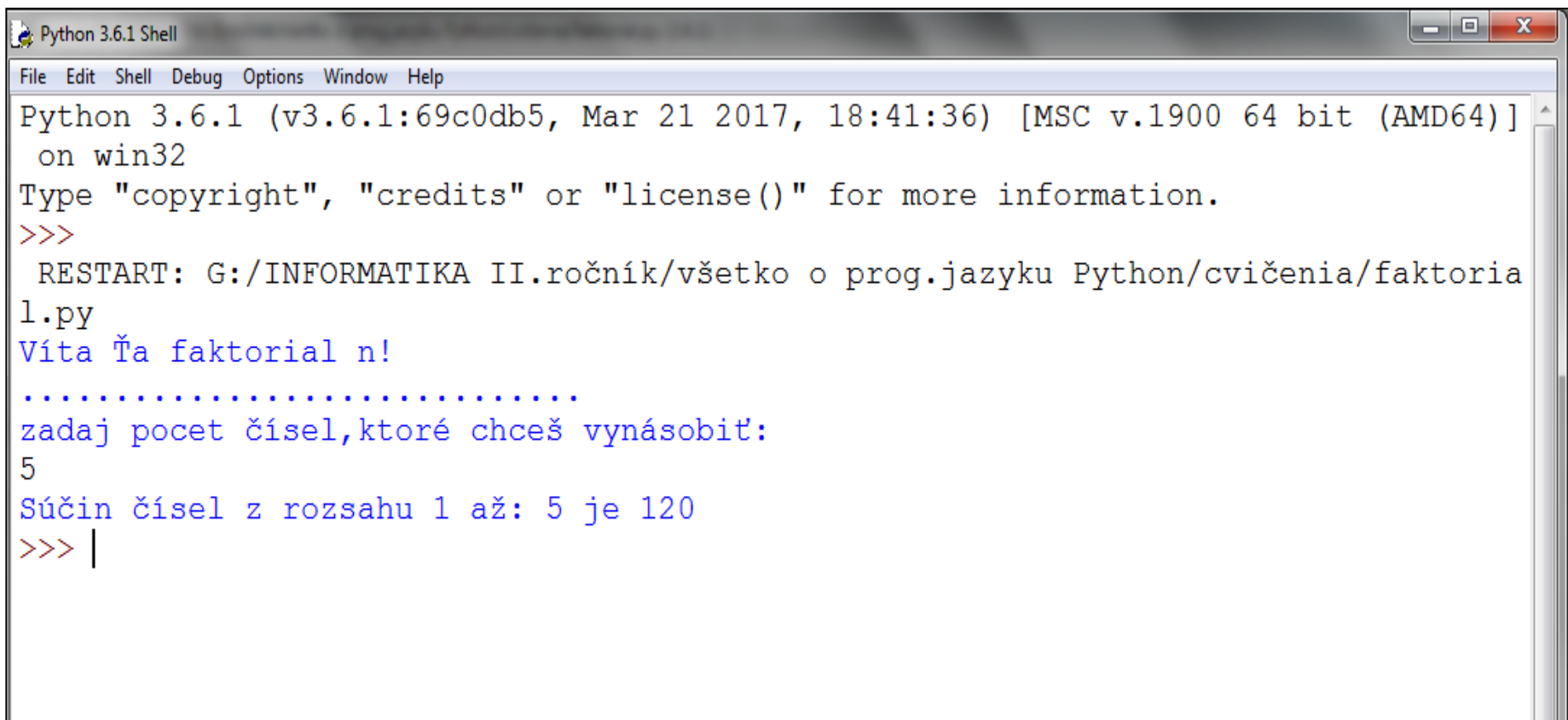
Pr.3.Vytvorte algoritmus na výpočet faktoriálu čísla



```
#Program na výpočet faktorialu -súčinu čísel zo zadaného rozsahu rozsahu  
#SucetVcykle.py  
print('Víta Ďa faktorial n!')  
print('.....')  
n = int(input('zadaj pocet čísel,ktoré chceš vynásobiť: '+\n'))  
sucin = 1  
for i in range(1, n+1):  
    sucin = sucin * i  
print('Súčin čísel z rozsahu 1 až: '+str(n)+' je', str(sucin))
```

#(C)ing.Pavel Drgo,27.jún 2017,12:48

shell



```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
  RESTART: G:/INFORMATIKA II.ročník/všetko o prog.jazyku Python/cvičenia/faktoria
l.py
Víta Ťa faktorial n!
.....
zadaj pocet čísel,ktoré chceš vynásobiť:
5
Súčin čísel z rozsahu 1 až: 5 je 120
>>> |
```

Použitie vnorených cyklov

```
#Program na použitie vnorených cyklov  
#trojuholnikZcisel.py  
print('Víta Ďa program vnorené cykly versia 1.0!')  
print('.....')  
pocet = int(input('Zadaj počet riadkov: '))  
for riadok in range(1, pocet+1):  
    for cislo in range(1, riadok+1):  
        print(cislo, end=' ')  
    print()  
 #(C)ing.Pavel Drgo,27.jún 2017,13:13
```

shell

```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:/INFORMATIKA II.ročník/všetko o prog.jazyku Python/cvičenia/trojuhohol
nikZcisel.py
Víta Ľa program vnorené cykly versia 1.0!
.....
Zadaj počet riadkov: 10
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
1 2 3 4 5 6 7
1 2 3 4 5 6 7 8
1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9 10
>>>
```

```
#Program na použitie vnorených cyklov  
#kamzik.py  
print('Víta Ďa program vnorené cykly versia 2.0!')  
print('.....')  
n = int(input('Zadaj počet riadkov piramidy: '))  
for i in range(n):  
    print(' '*(n-1-i) + '*'*(2*i+1))  
for i in range(n):  
    print(' '*i + '*'*(2*n-2*i-1))  
 #(C)ing.Pavel Drgo,27.jún 2017,13:22
```


Generátor náhodných čísel

- Aby sme mohli používať funkcie z tohto modulu generatora, musíme na začiatku

programu zapísať: **import random**

- V tomto module použijeme funkciu:

funkcia **random.randrange(spodok, vrch)**

- • funkcia má 1, 2 alebo 3 parametre s rovnakým významom ako range()
- • vyberie náhodnú hodnotu z tohto rozsahu
- • napr. `random.randrange(100)`-náhodné číslo od 0-100,
`random.randrange(10,100)`, náhodné číslo od 10-100
- `random.randrange(10,100,5)`-náhodné číslo od 10 do 100 s krokom 5

```
#Program na generovanie čísel zo zvolného rozsahu
#generatorZrozsahu.py
import random
print('Víta Ťa generátor náhodných čísel')
print('.....')
pocetCisel=(int(input('Zadaj počet čísel ,ktoré chceš
vygenerovať stlač ENTER'+'\n')))
dolnaHranica=(int(input('Zadaj spodnú hranicu generovaných
čísel a stlač ENTER'+'\n')))
hornaHranica=(int(input('Zadaj hornú hranicu generovaných
čísel a stlač ENTER'+'\n')))
for i in range(pocetCisel):
    nahoda=int(random.randrange(dolnaHranica,hornaHra
                                nica))
    print(str(i)+' . Generované číslo je:'+str(nahoda))
```

shell

```
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: G:/INFORMATIKA II.ročník/všetko o prog.jazyku Python/cvičenia/generato
rZrosahu.py
Víta Ľa generátor náhodných čísel
.....
Zadaj počet čísel ,ktoré chceš vygenerovať stlač ENTER
10
Zadaj spodnú hranicu generovaných čísel a stlač ENTER
1
Zadaj hornú hranicu generovaných čísel a stlač ENTER
1000
0. Generované číslo je:83
1. Generované číslo je:152
2. Generované číslo je:300
3. Generované číslo je:478
4. Generované číslo je:224
5. Generované číslo je:378
6. Generované číslo je:341
7. Generované číslo je:23
8. Generované číslo je:320
9. Generované číslo je:461
>>>
```

Funkcie

- Funkcia je pomenovaný blok príkazov (niekedy sa tomu hovorí aj podprogram).
- **def** meno_funkcie():
 prikaz1
 prikaz2
- ...

PR. **def** vypis():
 print('*****')
 print('*****')

Definícia tela funkcie končí vtedy, keď sa objaví riadok, ktorý už nie je odsunutý

Funkcie

Volanie funkcie

Volanie funkcie je taký zápis, ktorým sa začnú vykonávať príkazy z definície funkcie. Stačí zapísať meno funkcie so zátvorkami a funkcie sa spustí:

```
meno_funkcie()
```

Samozrejme, že funkciu môžeme zavolať až vtedy, keď už Python pozná jej definíciu.

```
def vypis():  
    print('*****')  
    print('*****')  
print('hello')  
vypis()  
print('* Python *')  
vypis()
```

funkcie

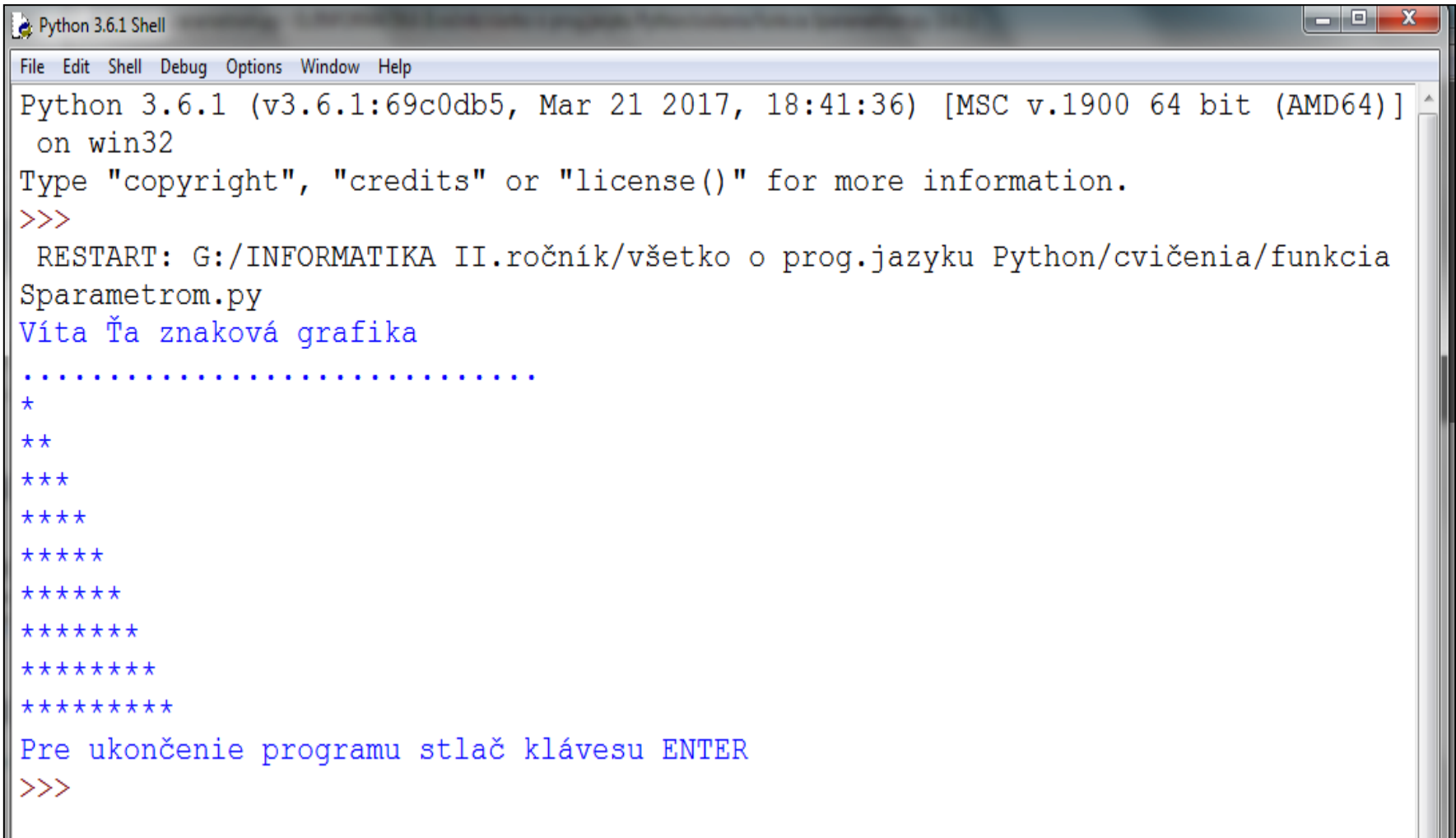
Parametre funkcie

- Parametrom funkcie je dočasná premenná, ktorá vzniká pri volaní funkcie a prostredníctvom ktorej, môžeme do funkcie poslať nejakú hodnotu.
- Parametre funkcií definujeme počas definovania funkcie v hlavičke funkcie
- Ak ich je viac, oddelujeme ich čiarkami:

```
def meno_funkcie(parameter):  
    prikaz1  
    prikaz2  
    ...
```

```
#Program na použitie funkcie s parametrom  
#funkciaSparametrom.py  
print('Víta Ľa znaková grafika')  
print('.....')  
def vypis_hviezdiciek(pocet):  
    print('*' * pocet)  
for i in range(1, 10):  
    vypis_hviezdiciek(i)  
input('Pre ukončenie programu stlač klávesu ENTER')  
 #(C)ing.Pavel Drgo,27jún 2017,12:21
```


shell



```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
  RESTART: G:/INFORMATIKA II.ročník/všetko o prog.jazyku Python/cvičenia/funkcia
  Sparametrom.py
  Víta Ťa znaková grafika
  .....
  *
  **
  ***
  ****
  *****
  *
  *****
  Pre ukončenie programu stlač klávesu ENTER
  >>>
```

Grafika v PYTHONU

[Material o grafike TKINTER](#)