

# OOP, Python

## A4B99RPH – Řešení problémů a hry

Tomáš Svoboda, [svobodat@fel.cvut.cz](mailto:svobodat@fel.cvut.cz)

katedra kybernetiky, centrum strojového vnímání

19. září 2012

01

# Plán

1. Poskytnout nezbytné základy z (objektového) programování těm, co o tom ještě neslyšeli.
2. Představit Python a ukázat příklady použití.
3. Živé (live) ukázky programovacích konstrukcí na přednášce.



# Proč Python?

- ▶ Neexistuje něco takového jako *Ten jeden ideální jazyk pro výuku programování.*
- ▶ V minulosti se používaly Basic, Fortran, Pascal, ....
- ▶ ..., nyní často Java, Delphi.
- ▶ Python je silný pro aplikace a zároveň snadný na začátek.
- ▶ Rychlý přechod mezi postupem řešení a vlastní implementací.
- ▶ Nevýhodou může být, že může existovat až příliš mnoho způsobů implementace.

Python dovolí rychle postoupit a zažít radost z fungujícího kódu.

01

# Proč Python?

- ▶ Neexistuje něco takového jako *Ten jeden ideální jazyk pro výuku programování.*
- ▶ V minulosti se používaly Basic, Fortran, Pascal, ....
- ▶ ..., nyní často Java, Delphi.
- ▶ Python je silný pro aplikace a zároveň snadný na začátek.
- ▶ Rychlý přechod mezi postupem řešení a vlastní implementací.
- ▶ Nevýhodou může být, že může existovat až příliš mnoho způsobů implementace.

Python dovolí rychle postoupit a zažít radost z fungujícího kódu.

01

# Proč Python?

- ▶ Neexistuje něco takového jako *Ten jeden ideální jazyk pro výuku programování.*
- ▶ V minulosti se používaly Basic, Fortran, Pascal, ....
- ▶ ..., nyní často Java, Delphi.
- ▶ Python je silný pro aplikace a zároveň snadný na začátek.
- ▶ Rychlý přechod mezi postupem řešení a vlastní implementací.
- ▶ Nevýhodou může být, že může existovat až příliš mnoho způsobů implementace.

Python dovolí rychle postoupit a zažít radost z fungujícího kódu.

01

# Proč OOP?

- ▶ vhodný přístup pro mnoho reálných problémů
- ▶ mnoho nyní používaných jazyků je objektových (např. Java)
- ▶ pro naše úlohy to budeme potřebovat



# Python – ne tak obvyklý úvod

- ▶ předpokládáme intenzivní samostudium
- ▶ vybrané kapitoly  
z [Wentworth2012, Summerfield2010, Kubias2008]
  - ▶ jen to, co budeme potřebovat:
  - ▶ pole (seznamy), funkce, třídy, *objekty*
  - ▶ zbytek samostudium, a dotazy/diskuse na cvičení

Obrázky a příklady jsou převážně z [Downey2009] (nově [Wentworth2012]), či jsou touto knihou silně inspirovány. Přístup „objekty nejdřív“ je, kromě úloh z toho předmětu, také volně inspirován knihou [Pecinovsky2010]

01

# Python – ne tak obvyklý úvod

- ▶ předpokládáme intenzivní samostudium
- ▶ vybrané kapitoly  
z [Wentworth2012, Summerfield2010, Kubias2008]
- ▶ jen to, co budeme potřebovat:
- ▶ pole (seznamy), funkce, třídy, *objekty*
- ▶ zbytek samostudium, a dotazy/diskuse na cvičení

Obrázky a příklady jsou převážně z [Downey2009] (nově [Wentworth2012]), či jsou touto knihou silně inspirovány. Přístup „objekty nejdřív“ je, kromě úloh z toho předmětu, také volně inspirován knihou [Pecinovsky2010]





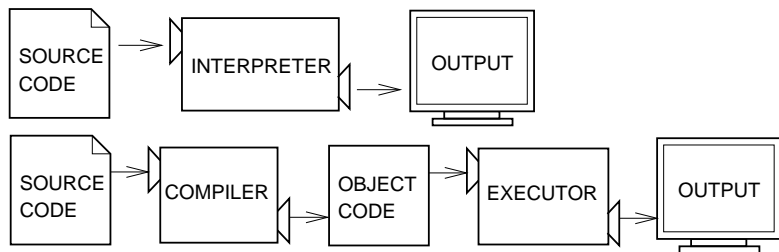
# Python – ne tak obvyklý úvod

- ▶ předpokládáme intenzivní samostudium
- ▶ vybrané kapitoly  
z [Wentworth2012, Summerfield2010, Kubias2008]
- ▶ jen to, co budeme potřebovat:
- ▶ pole (seznamy), funkce, třídy, *objekty*
- ▶ zbytek samostudium, a dotazy/diskuse na cvičení

Obrázky a příklady jsou převážně z [Downey2009] (nově [Wentworth2012]), či jsou touto knihou silně inspirovány. Přístup „objekty nejdřív“ je, kromě úloh z toho předmětu, také volně inspirován knihou [Pecinovsky2010]



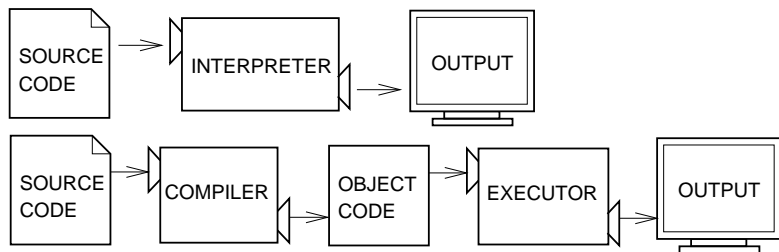
# Python – interpretovaný jazyk



Python:

- ▶ objektově orientované programování
- ▶ strukturované programování (imperativní)

# Python – interpretovaný jazyk



## Python:

- ▶ objektově orientované programování
- ▶ strukturované programování (imperativní)

# Python, pár příkladů

Live ukázky

01

# Otázky?

- ▶ pokud si vzpomenete později, použijte diskusní fórum
- ▶ nebo čas na cvičení
- ▶ nebo si svoji otázku poznamenejte a položte ji na další přednášce

**Otázky jsou velmi důležité.**



# Literatura I

- [Downey2009] Allen B. Downey.  
*Python for software design.*  
Cambridge University Press, 2009.  
Also on-line at  
<http://www.greenteapress.com/thinkpython/thinkpython.html>.
- [Kubias2008] Jeffrey Elkner, Allen B. Downey, and Chris Meyers.  
*Učíme se programovat v Jazyce Python (How To Think Like a Computer Scientist - Learning with Python).*  
howto.py.cz, 2 edition, březen 2008.  
České vydání překlad Jaroslav Kubias. <http://howto.py.cz/index.htm>.
- [Pecinovsky2010] Rudolf Pecinovský.  
*Naučte se myslet a programovat objektově.*  
Computer Press, 1st edition, 2010.
- [Summerfield2010] Mark Summerfield.  
*Python 3: Výukový kurz.*  
Computer Press, 1st edition, 2010.



# Literatura II

[Wentworth2012] Peter Wentworth, Jeffrey Elkner, Allen B. Downey, and Chris Meyers.

*How To Think Like a Computer Scientist - Learning with Python 3 (RLE).*

<http://openbookproject.net/thinkcs/python/english3e/>, 3 edition, 2012.  
based on Downey 2009.

