

# Programování 2

---

## 1. cvičení, 17-2-202

---

tags: Programování 2, čtvrtek 1, čtvrtek 2

### Obsah:

- co budeme dělat v tomto semestru
- co od sebe očekáváme a pravidla hry

## Co budeme dělat

---

### NMN112 Programování II

Přednáška bude věnována zejména problematice algoritmického návrhu programů – tedy seznámení se základními algoritmy, datovými strukturami a programovacími technikami s ohledem na efektivitu vytvářených programů. Na cvičeních vedle toho doplníme a procvičíme znalosti programovacího jazyka Python.

Na těchto čtvrtěčnících cvičeních budeme pokračovat tam, kde jsme začli v minulém semestru - budeme se učit psát dobrý kód v Pythonu. Kromě psaní správného, čitelného a kompaktního kódu se budeme věnovat i

- efektivitě kódu - už nebude stačit syntakticky správný kód, který dělá to, co má: budu chtít, aby váš kód dělal, co má, s rozumným využitím výpočetních prostředků - paměti a času.
- Budeme se učit implementovat v Pythonu algoritmy z přednášek a teoretických cvičení. Budete psát hodně kódu, abyste se naučili psát kód rychle a rychle ho také odladit.
- Budete také mít spoustu možností vymýšlet si své vlastní algoritmy pro různé úlohy.

Přibližný program - zatím bez uspořádání podle týdnů:

1. Úvod, podmínky k zápočtu, ReCodEx. Vývojová prostředí.
2. Jednoduché problémy, návrh a ladění algoritmů
3. Třídění - různé algoritmy, halda (heap) a její implementace
4. Permutace
5. Hry a optimální strategie
6. Rekurze, memoizace, dynamické programování
7. Grafové algoritmy, zásobníky a fronty

K tomuto "hlavnímu" programu pak přidáme "podprogramy" -- u každého cvičení se naučíme také něco nového z Pythonu, a budeme se učit číst kód.

### Pomoc

Budeme dělat jednoduché věci, ale programování je spojeno s častými pocity frustrace, když vám nebude fungovat něco, co by podle vás určitě fungovat mělo.

Základní postup v takovémto případě je **zeptat se lámanou angličtinou Googlu**. Zpravidla rychle najdete kvalifikovanou odpověď.

**Klidně se ptejte i mě**, v průběhu cvičení, nebo e-mailem ([peter.kvasnicka@mff.cuni.cz](mailto:peter.kvasnicka@mff.cuni.cz)), nebo mi zavolejte a dohodneme se na konzultaci (605 386 052). Pohotově se zeptat je schopnost, která vám může ušetřit kópu času a zbytečné frustrace, takže je dobré ji kultivovat. Žádná otázka není hloupá.

# Podmínky zápočtu

Budete dostávat domácí úkoly a odevzdávat je přes *ReCodEx*.

Tak jako v minulém semestru se budu snažit zadávat tři úlohy týdně, dvě "povinné" a jednu bonusovou. Body ode všech úloh jsou rovnocenné, ale body od bonusových úloh nezvyšují celkovou kvótu bodů za semestr.

Pro zápočet budu požadovat **70% správných odevzdaných domácích úkolů**.

V tomto semestru ale kromě domácích úkolů budete k zápočtu potřebovat také **zápočtový program**. Přesné podmínky si domluvíme později v průběhu semestru.

## Instalace Pythonu

Na počítačích v učebně byste měli najít vše potřebné. Pokud ne, *stěžujte si*.

Tady máme vícero možností a nechám na váš výběr, kterou si zvolíte.

### 1. Základní distribuce Pythonu

Stáhněte si instalátor pro svůj systém tady: <https://www.python.org/downloads/>.

Zvolte si nejnovější verzi 3.9. Součástí je vlastní interpret a jednoduché IDE *Idle*. S tímto klidně vystačíte pro celý kurz.

### 2. Anaconda

Toto je velká distribuce, která obsahuje rozsáhlou podporu pro využití Pythonu ke zpracování dat, strojové učení a pod. Stáhněte si ji tady: <https://www.anaconda.com/products/individual> a zabere vám docela hodně místa na disku. Součástí je i vyspělé IDE pro vývoj v Pythonu - *Spyder*.

### 3. Google Colab notebooky

Nemusíte nic instalovat, stačí jít na [colab.google.com](https://colab.google.com) a začít psát kód do notebooku.

## IDE pro Python

Existuje několik programovacích editorů a vývojových prostředí pro Python, například *PyCharm*, *VSCode*, *Atom*, *Sublime Text*, atd.

V tomto semestru nám už *Idle* nebude stačit, budeme psát složitější kódy a budeme dělat těžší chyby, takže se naučíme i nějaké techniky ladění kódu. Na přednáškách budu používat *-PyCharm* a *VSCode*, abyste si na obě prostředí zvykli.

## ReCodEx: systém pro kontrolu domácích úloh

Zaregistrujte se v ReCodExu, <https://recodex.mff.cuni.cz/>. Můžete použít svoji přístupové údaje do SISu. Pak se prosím zaregistrujte do skupiny pro toto cvičení, můžete tak učinit volbou "SIS integration".

V ReCodExu najdete své domácí úkoly a budete je tady i odevzdávat.

V minulém semestru jsme často s ReCodExem zápolili, takže pokud se vám zdá, že váš kód je správný, ale ReCodEx je k vám nespravedlivý, nahrajte řešení a napište buď komentář k úloze anebo mi pošlete e-mail.

**První úlohy najdete v ReCodExu koncem týdne.**

# Co jsme se naučili v předchozím semestru

---

Samozřejmě jste udělali první kroky v Pythonu - tedy alespoň ti z vás, kteří je neudělali už dříve. Získali jsme také ale několik magických schopností:

- práce se seznamy a znakovými řetězci
- třídění
- binární vyhledávání
- rekurzivní funkce
- generátory
- funkce, vracející funkce
- (velice nekompletní) třídy

V tomto semestru tento seznam rozšíříme o nové skilly.