

Lekce 10: Úvod do ArcPy

Python pro GIS - První kroky s automatizací ArcGIS Pro

Vojtěch Barták, FŽP ČZU Praha

2025-10-31

Table of contents

1	Cíle lekce	4
2	Co je ArcPy?	4
2.1	Definice	4
2.2	Další Python knihovny pro práci s ArcGIS	5
2.2.1	ArcGIS API for Python	5
2.2.2	Další knihovny	5
3	Tři způsoby spuštění Pythonu v ArcGIS Pro	5
3.1	Python Window	6
3.1.1	Kde ho najít	6
3.1.2	Charakteristika	6
3.1.3	Kdy použít	6
3.1.4	Demo - první příkaz	6
3.2	Notebook	7
3.2.1	Kde ho vytvořit	7
3.2.2	Charakteristika	7
3.2.3	Kdy použít	7
3.2.4	Demo - struktura notebooku	7
3.3	3. Python Script (.py soubor)	8
3.3.1	Kde ho vytvořit	8
3.3.2	Jak spustit	8
3.3.3	Charakteristika	10
3.3.4	Kdy použít	11
3.3.5	Demo - první skript	11
4	První kroky s ArcPy	12
4.1	Import modulu	12

4.2	Základní informace o instalaci	12
4.3	Nastavení workspace	13
4.4	Přepisování výstupů	14
4.5	Licencování rozšíření	14
4.5.1	Kontrola a aktivace licence	14
5	První praktický příklad - Buffer	15
5.1	Úloha	15
5.2	Řešení ve třech krocích	15
5.2.1	Krok 1: Python Window (rychlý test)	15
5.2.2	Krok 2: Notebook (s dokumentací)	16
5.2.3	Krok 3: Script (pro opakované spouštění)	16
5.3	Syntaxe nástroje Buffer	17
6	Praktické cvičení	18
6.1	Příprava dat	18
6.2	Úkol 1: Buffer v Python Window (5 min)	18
6.3	Úkol 2: Notebook s dokumentací (10 min)	18
6.4	Úkol 3: Script pro batch processing (volitelné)	19
7	Shrnutí	19
7.1	Co jsme se naučili	19
7.2	Kdy použít který způsob?	19
7.3	Co bude příště?	20
8	Domácí úkol	20
8.1	Varianta A (základní)	20
8.2	Varianta B (střední)	20
8.3	Varianta C (pokročilá)	21
8.4	Varianta D (pokročilá)	21
8.5	Varianta E (výzva pro experty)	21
9	Cheatsheet	22
10	Poznámky pro vyučujícího	24
10.1	Běžné chyby studentů	24
10.2	Časový plán (90 min)	24
10.3	Klíčové momenty	25
10.3.1	Python Window vs. Notebook vs. Script (10-30 min):	25
10.3.2	Nastavení workspace (30-45 min):	25
10.3.3	Licencování (30-45 min):	25
10.3.4	Buffer - společně (45-70 min):	25
10.4	Rizika	26
10.5	Tipy	26

10.6 Materiály k přípravě	26
-------------------------------------	----

1 Cíle lekce

Po absolvování této lekce budete umět:

- Vysvětlit, co je ArcPy a k čemu slouží
- Spustit Python kód třemi způsoby v ArcGIS Pro (Python Window, Notebook, Script)
- Importovat modul `arcpy` a nastavit základní prostředí
- Vytvořit první jednoduchý geoprocessingový skript (buffer)
- Pracovat s licencováním rozšíření (Spatial Analyst)
- Rozhodnout, kdy použít který způsob spuštění Pythonu

Časová dotace: 90 minut

2 Co je ArcPy?

2.1 Definice

ArcPy je Python knihovna (balíček), která zprostředkovává funkcionalitu ArcGIS Pro do jazyka Python.

Díky ArcPy můžete:

- **Automatizovat** opakující se úlohy
- **Zpracovat** velké množství dat (batch processing)
- **Kombinovat** nástroje ArcGIS s možnostmi Pythonu
- **Vytvořit** vlastní nástroje s parametry
- **Integrovat** GIS do složitějších workflow

Vztah k Model Builderu

Model Builder je výborný pro vizuální návrh workflow, ale má limity:

- Těžko se v něm píše **podmínky** (if-else)
- Nelze vytvořit **vlastní funkce**
- **Opakování** (cykly) je omezené
- Složité **logické operace** jsou nepohodlné

Python + ArcPy = plná kontrola nad GIS analýzami!

2.2 Další Python knihovny pro práci s ArcGIS

ArcPy není jediná Python knihovna pro práci s geografickými daty v ekosystému Esri. Existují další knihovny, které rozšiřují možnosti práce s ArcGIS:

2.2.1 ArcGIS API for Python

ArcGIS API for Python (modul `arcgis`) je moderní knihovna pro práci s **ArcGIS Online** a **ArcGIS Enterprise**.

Typické použití: - Správa webových map a služeb - Publikování dat na ArcGIS Online - Práce s portálem (vytváření uživatelů, skupin) - Stahování dat z webových služeb - Prostorová analýza v cloudovém prostředí - Jupyter Notebooks s interaktivními mapami

Hlavní rozdíl oproti ArcPy: - **ArcPy** = lokální analýzy v ArcGIS Pro - **arcgis** = webové služby, cloudové analýzy, správa portálu

2.2.2 Další knihovny

Kromě ArcPy můžete využít i další populární Python knihovny pro GIS či obecně pro analýzu dat:

- **pandas** - práce s tabulkovými daty, statistika
- **numpy** - numerické výpočty, práce s rastry
- **matplotlib / seaborn** - vizualizace dat, grafy
- **scikit-learn** - strojové učení
- **geopandas** - prostorová data v pandas stylu (alternativa k ArcPy)

💡 Kdy použít kterou knihovnu?

V tomto kurzu: - Zaměřujeme se na **ArcPy** pro lokální analýzy v ArcGIS Pro - ArcPy je základ pro automatizaci GIS úloh

V praxi: - Často kombinujete **ArcPy** (GIS nástroje) + **pandas** (zpracování tabulek) + **matplotlib** (grafy) - Pro webové aplikace a cloudové služby používáte **ArcGIS API for Python**

3 Tři způsoby spuštění Pythonu v ArcGIS Pro

V ArcGIS Pro můžete Python spouštět **třemi způsoby**. Každý má své výhody.

3.1 Python Window

3.1.1 Kde ho najít

View → Python Window nebo Analysis → Python → Python Window

[videos/python_window.mp4](#)

3.1.2 Charakteristika

- **Interaktivní** - píšete příkaz, okamžitě vidíte výsledek
- **Rychlé testování** - bez ukládání souborů
- **Autocomplete** - našeptávání funkcí
- **Historie** - šipky nahoru/dolů pro předchozí příkazy

3.1.3 Kdy použít

Rychlé testy (“Jak se jmenuje ta funkce?”)

Zjištění syntaxe nástroje

Explorativní práce s daty

Kontrola vlastností dat

Složité skripty (> 10 řádků)

Opakované spouštění

Sdílení kódu

3.1.4 Demo - první příkaz

```
import arcpy
info = arcpy.GetInstallInfo()
print(f"Verze ArcGIS Pro: {info['Version']}")
```

💡 Python Window je váš přítel!

Když si nejste jistí syntaxí, vždy to nejdřív zkuste v Python Window.

3.2 Notebook

3.2.1 Kde ho vytvořit

Insert → New Notebook nebo Analysis → Python → Python Notebook

[videos/python_notebook.mp4](#)

3.2.2 Charakteristika

- **Buňky** - Code buňky (Python) + Markdown buňky (text, obrázky)
- **Interaktivní** - můžete spouštět buňku po buňce
- **Dokumentace** - vysvětlení přímo v notebooku
- **Vizualizace** - grafy, mapy, výsledky
- **Sdílení** - export do HTML, PDF

3.2.3 Kdy použít

Explorativní datová analýza

Dokumentace postupu (kombinace kódu + textu)

Výuka a sdílení znalostí

Reporty s grafy a mapami

Automatizace (nelze snadno spouštět z příkazové řádky)

Produkční skripty

3.2.4 Demo - struktura notebooku

Markdown buňka:

```
# Analýza populace měst
```

```
Tento notebook zpracovává data o městech a vytváří buffer.
```

Code buňka:

```
import arcpy  
arcpy.env.workspace = r"C:\Data\arccr500\AdministrativniCleneni_v13.gdb"
```

Další Code buňka:

```
arcpy.analysis.Buffer("OkresyBody", "okresy_buffer", "1000 Meters")
print("Buffer vytvořen!")
```

Jupyter Notebooks

Notebooky v ArcGIS Pro jsou založené na **Jupyter Notebooks** - velmi populárním nástroji v datové vědě a GIS komunitě.

Tento nástroj máte spolu s ArcGIS Pro rovněž k dispozici: - Start menu → **ArcGIS** → **Jupyter Notebook**

3.3 3. Python Script (.py soubor)

3.3.1 Kde ho vytvořit

Jakýkoli textový editor:

- **IDLE** (Python editor dodávaný s Pythonem)
- **Poznámkový blok** (Notepad) - funguje, ale nepraktické
- **Notepad++** (doporučeno pro začátečníky)
- **VS Code** (doporučeno pro pokročilejší)
- **PyCharm** (profesionální IDE)

3.3.2 Jak spustit

POZOR: Skript .py **není viditelný** v Catalog Pane ani Geoprocessing Pane! Tyto nástroje zobrazují pouze **Python Toolboxy** (.pyt, .tbx).

Pro spuštění běžného .py skriptu máte tyto možnosti:

3.3.2.1 A) IDLE (doporučeno pro začátečníky)

Způsob 1: Spuštění IDLE přímo

1. Najděte **Python Command Prompt** z ArcGIS Pro:

- Start menu → **ArcGIS** → **Python Command Prompt**

2. V otevřeném okně napište:


```
idle
```

nebo

```
python -m idlelib
```

3. IDLE se otevře → **File** → **Open** → vyberte váš .py soubor
4. **Run** → **Run Module** (F5)

Způsob 2: Přímé spuštění skriptu v IDLE

1. Právý klik na váš .py soubor
2. **Edit with IDLE** (pokud je tato možnost dostupná)

i Pokud “Edit with IDLE” není k dispozici

IDLE nemusí být ve Windows registrováno pro .py soubory. V tom případě použijte **Způsob 1** nebo **Způsob 3**.

Způsob 3: Vytvoření zástupce na ploše

Pro pohodlnější práci vytvořte zástupce:

1. Právý klik na plochu → **New** → **Shortcut**
2. Do cesty zadejte:

```
"C:\Program Files\ArcGIS\Pro\bin\Python\envs\arcgispro-py3\python.exe" -m idlelib
```

3. Pojmenujte: **IDLE (ArcGIS Pro)**
4. Klikněte na zástupce → otevře se IDLE s ArcPy

! Musíte použít Python z ArcGIS Pro!

NEFUNGUJE: Standardní Python z python.org (nemá ArcPy)

FUNGUJE: Python nainstalovaný s ArcGIS Pro

Cesta k Pythonu ArcGIS Pro:

```
C:\Program Files\ArcGIS\Pro\bin\Python\envs\arcgispro-py3\python.exe
```

Tento Python obsahuje knihovnu **arcpy**.

Ověření: V IDLE po spuštění zkuste:

```
>>> import arcpy
>>> print(arcpy.GetInstallInfo()['Version'])
3.2.0 # Měla by se zobrazit verze ArcGIS Pro
```

3.3.2.2 B) Příkazová řádka (Command Prompt)

```
# Navigujte do složky se skriptem
cd C:\MojeSkripty

# Spusťte pomocí Python z ArcGIS Pro
"C:\Program Files\ArcGIS\Pro\bin\Python\envs\arcgispro-py3\python.exe" prvni_buffer.py
```

Zkrácená verze (pokud přidáte ArcGIS Python do PATH):

```
python prvni_buffer.py
```

Jak přidat ArcGIS Python do PATH

1. Zkopírujte: C:\Program Files\ArcGIS\Pro\bin\Python\envs\arcgispro-py3
2. **Tento PC** → pravý klik → **Properties**
3. **Advanced system settings** → **Environment Variables**
4. V **System variables** najděte **Path** → **Edit**
5. **New** → vložte zkopírovanou cestu
6. **OK, OK, OK**

Po restartu příkazové řádky můžete použít pouze **python** místo celé cesty.

3.3.2.3 C) Spuštění z editoru (VS Code, PyCharm)

V pokročilých editorech můžete nastavit **Python Interpreter** na ArcGIS Pro verzi:

VS Code: 1. Ctrl + Shift + P → “Python: Select Interpreter” 2. Vyberte: C:\Program Files\ArcGIS\Pro\bin\Python\envs\arcgispro-py3\python.exe 3. Nyní můžete spustit F5

3.3.3 Charakteristika

- **Opakovatelné** - uložený soubor lze spustit znovu
- **Verzovatelné** - Git, správa verzí

- **Automatizace** - spouštění z batch souborů, Windows Task Scheduler
- **Sdílení** - jednoduché sdílení .py souboru
- **Flexibilní** - můžete spouštět odkudkoli (nejen z ArcGIS Pro)

3.3.4 Kdy použít

Produkční skripty (opakované spouštění)
 Složitější logika (> 50 řádků)
 Automatizace (naplánované úlohy)
 Integrace do větších systémů
 Spolupráce (verzování pomocí Git)
 Batch processing mimo ArcGIS Pro

Rychlé testování (pomalé ukládání + spouštění)
 Interaktivní explorativní práce

3.3.5 Demo - první skript

Vytvořte soubor `prvni_buffer.py` v **IDLE**:

```
"""
Skript pro vytvoření bufferu kolem silnic
Autor: Váš tým
Datum: 2025
"""

import arcpy

# Nastavení prostředí
arcpy.env.workspace = r"C:\Data\Projekt.gdb"
arcpy.env.overwriteOutput = True

# Parametry
vstup = "silnice"
vystup = "silnice_buffer"
vzdalenost = "100 Meters"

# Vytvoření bufferu
print(f"Vytvářím buffer pro {vstup}...")
arcpy.analysis.Buffer(vstup, vystup, vzdalenost)
print(f"Buffer vytvořen: {vystup}")
```

```
print("\nHotovo! Zkontrolujte výsledek v ArcGIS Pro.")
```

Spuštění: 1. Uložte soubor (Ctrl + S) 2. V IDLE: Run → Run Module (F5) 3. Výsledek se zobrazí v konzoli IDLE 4. Otevřete ArcGIS Pro a zkontrolujte vrstvu **silnice_buffer**

⚠ Python Window vs. Python Script

Hlavní rozdíl:

- **Python Window** běží **uvnitř** ArcGIS Pro → výsledky vidíte okamžitě na mapě
- **Python Script** (.py) běží **mimo** ArcGIS Pro → musíte otevřít ArcGIS Pro a zkontrolovat výsledky ručně

V praxi: 1. Testujte v **Python Window** (rychlá zpětná vazba) 2. Finální verzi uložte jako **.py skript** (opakovatelnost)

4 První kroky s ArcPy

4.1 Import modulu

Každý Python skript pracující s ArcGIS musí začít:

```
import arcpy
```

! Kde funguje import arcpy?

`import arcpy` funguje **jen** s Pythonem nainstalovaným s ArcGIS Pro!
Pokud spustíte Python mimo ArcGIS prostředí, dostanete chybu: `ModuleNotFoundError: No module named 'arcpy'`

4.2 Základní informace o instalaci

```
import arcpy

# Zjištění informací o ArcGIS Pro
info = arcpy.GetInstallInfo()

print(f"Produkt: {info['ProductName']}")
print(f"Verze: {info['Version']}")
print(f"Instalační adresář: {info['InstallDir']}")
print(f"Python: {info['PythonVersion']}")
```

Výsledek:

```
Produkt: ArcGISPro
Verze: 3.2.0
Instalační adresář: C:\Program Files\ArcGIS\Pro
Python: 3.11.8
```

4.3 Nastavení workspace

Workspace = pracovní adresář, kde ArcPy hledá a ukládá data.

```
import arcpy

# Nastavení geodatabáze jako workspace
arcpy.env.workspace = r"C:\Data\MojeProjekt.gdb"

# Nastavení složky jako workspace
arcpy.env.workspace = r"C:\Data\Shapefiles"
```

💡 Proč r před cestou?

`r"C:\Data\..."` - **raw string** = Python nebere \ jako escape znak.

Alternativy:

```
arcpy.env.workspace = r"C:\Data\Projekt.gdb" # raw string (doporučeno)
arcpy.env.workspace = "C:/Data/Projekt.gdb" # lomítka (funguje taky)
arcpy.env.workspace = "C:\\Data\\Projekt.gdb" # dvojité zpětné lomítko
```

Doporučujeme **raw string** (`r"..."`).

4.4 Přepisování výstupů

Pokud výstupní soubor už existuje, ArcPy **vrátí chybu**.

Řešení: Povolit přepisování:

```
arcpy.env.overwriteOutput = True
```

nebo

```
arcpy.env.overwriteOutput = 1
```

4.5 Licencování rozšíření

Některé nástroje vyžadují **rozšíření** (extensions):

- **Spatial Analyst** - rastrové analýzy
- **3D Analyst** - 3D analýzy
- **Network Analyst** - síťové analýzy

4.5.1 Kontrola a aktivace licence

```
import arcpy

# Kontrola dostupnosti
status = arcpy.CheckExtension("Spatial")

if status == "Available":
    # Aktivace licence
    arcpy.CheckOutExtension("Spatial")
    print("Spatial Analyst aktivován")

    # ... vaše analýzy ...

    # Vrácení licence (uvolnění)
    arcpy.CheckInExtension("Spatial")
else:
    print(f"Spatial Analyst není dostupný: {status}")
```

Možné stavy:

- "Available" - dostupná
- "Unavailable" - nedostupná
- "NotLicensed" - není licencovaná
- "Failed" - chyba při aktivaci

i Kdy je třeba CheckOutExtension?

Jen když používáte nástroje z rozšíření!

Základní nástroje (Buffer, Clip, ...) **nepotřebují** rozšíření.

5 První praktický příklad - Buffer

Nyní zkombinujeme vše, co jsme se naučili.

5.1 Úloha

Vytvořte buffer 100 metrů kolem vrstvy silnic.

5.2 Řešení ve třech krocích

5.2.1 Krok 1: Python Window (rychlý test)

Otevřete Python Window a napište:

```
import arcpy
arcpy.env.workspace = r"C:\Data\Projekt.gdb"
arcpy.analysis.Buffer("silnice", "silnice_buffer", "100 Meters")
print("Buffer vytvořen!")
```

Výhoda: Okamžitá zpětná vazba - vidíte výsledek na mapě.

5.2.2 Krok 2: Notebook (s dokumentací)

Vytvořte nový Notebook a rozdělte do buněk:

Buňka 1 (Markdown):

```
# Analýza dopravní sítě

Cíl: Vytvořit ochranné pásmo 100 m kolem silnic.
```

Buňka 2 (Code):

```
import arcpy
arcpy.env.workspace = r"C:\Data\Projekt.gdb"
arcpy.env.overwriteOutput = True
```

Buňka 3 (Code):

```
# Parametry
vstup = "silnice"
vystup = "silnice_buffer"
vzdalenost = "100 Meters"

# Vytvoření bufferu
arcpy.analysis.Buffer(vstup, vystup, vzdalenost)
print(f"Buffer vytvořen: {vystup}")
```

Buňka 4 (Markdown):

```
## Výsledek

Buffer byl úspěšně vytvořen. Další krok: průnik s dalšími vrstvami.
```

Výhoda: Dokumentace + kód dohromady.

5.2.3 Krok 3: Script (pro opakované spouštění)

Vytvořte soubor `buffer_silnice.py`:


```

"""
Skript pro vytvoření ochranného pásma kolem silnic
Autor: GIS tým
Datum: 2025-10-31
"""

import arcpy

def vytvor_buffer(vstup, vystup, vzdalenost):
    """
    Vytvoří buffer kolem vstupní vrstvy.

    Parametry:
        vstup (str): Název vstupní vrstvy
        vystup (str): Název výstupní vrstvy
        vzdalenost (str): Vzdálenost bufferu (např. "100 Meters")
    """
    print(f"Vytvářím buffer pro {vstup}...")
    arcpy.analysis.Buffer(vstup, vystup, vzdalenost)
    print(f" Buffer vytvořen: {vystup}")

# Hlavní část skriptu
if __name__ == "__main__":
    # Nastavení prostředí
    arcpy.env.workspace = r"C:\Data\Projekt.gdb"
    arcpy.env.overwriteOutput = True

    # Spuštění
    vytvor_buffer("silnice", "silnice_buffer", "100 Meters")

```

Výhoda: Opakovatelné, sdílitelné, verzovatelné.

5.3 Syntaxe nástroje Buffer

```

arcpy.analysis.Buffer(
    in_features,          # Vstupní vrstva
    out_feature_class,    # Výstupní vrstva
    buffer_distance_or_field, # Vzdálenost
    line_side="FULL",     # Strana (FULL/LEFT/RIGHT)
    line_end_type="ROUND", # Zakončení (ROUND/FLAT)

```

```
dissolve_option="NONE" # Rozpuštění (NONE/ALL/LIST)
)
```

💡 Jak zjistit syntaxi nástroje?

1. **ArcGIS Pro Help** - dokumentace online
2. **Python Window** - autocomplete (Ctrl + Space)
3. **Geoprocessing pane** - spustit nástroj manuálně, pak **Copy Python Command**

[videos/copy_python.mp4](#)

6 Praktické cvičení

6.1 Příprava dat

Stáhněte si [zde](#):

- ArcCR500_v33.gdb - geodatabáze s vrstvami
- VodniPlochy - bodová vrstva rybníků, jezer a nádrží
- Lesy - polygonová vrstva lesů

6.2 Úkol 1: Buffer v Python Window (5 min)

V Python Window vytvořte:

1. Buffer 50 metrů kolem VodniPlochy
2. Výstup pojmenujte VodniPlochy_buffer50

6.3 Úkol 2: Notebook s dokumentací (10 min)

Vytvořte Notebook, který:

1. **Markdown buňka:** Nadpis “Analýza ochranných pásem vodních ploch”
2. **Code buňka:** Import a nastavení workspace
3. **Code buňka:** Buffer 50 m
4. **Code buňka:** Buffer 100 m
5. **Markdown buňka:** Shrnutí (kolik bufferů bylo vytvořeno)

6.4 Úkol 3: Script pro batch processing (volitelné)

Vytvořte skript, který:

1. Vytvoří buffery **tří různých** vzdáleností: 50, 100, 200 metrů
2. Použije **cyklus** for
3. Výstupy pojmenuje: VodniPlochy_buffer50, VodniPlochy_buffer100, ...

Hint:

```
vzdalenosti = [50, 100, 200]
for v in vzdalenosti:
    vystup = f"VodniPlochy_buffer{v}"
    # ... váš kód
```

7 Shrnutí

7.1 Co jsme se naučili

ArcPy = Python knihovna pro ArcGIS Pro

Tři způsoby spuštění: Python Window, Notebook, Script

import arcpy - základ každého GIS skriptu

Nastavení prostředí - workspace, overwriteOutput

Licencování - CheckOutExtension pro rozšíření

První nástroj - arcpy.analysis.Buffer()

7.2 Kdy použít který způsob?

Python Window:

- Rychlé testy
- Zjištění syntaxe
- Explorativní práce

Notebook:

- Dokumentovaná analýza
- Sdílení postupu
- Výuka

Script (.py):

- Produkční skripty
- Automatizace
- Složitější logika

7.3 Co bude přístě?

V příští lekci (Lekce 11):

- **Správa dat** - listing, describe, exists
 - **Procházení geodatabáze** - ListFeatureClasses, ListFields
 - **Batch processing** - zpracování více souborů najednou
-

8 Domácí úkol

8.1 Varianta A (základní)

Pomocí **Python Window**:

1. Vytvořte buffer 200 metrů kolem vrstvy **lesy**
2. Výstup pojmenujte **lesy_buffer200**
3. Vypište informaci o úspěšném vytvoření

8.2 Varianta B (střední)

Vytvořte **Notebook**, který:

1. Vytvoří tři buffery kolem **VodniPlochy**: 25, 50, 75 metrů
2. Pro každý buffer vypíše zprávu o vytvoření
3. Obsahuje Markdown buňky s dokumentací (nadpis, popis úlohy, shrnutí)
4. Exportujte Notebook jako PDF, HTML nebo .ipynb soubor.

8.3 Varianta C (pokročilá)

Vytvořte **Python skript** `buffer_batch.py`, který:

1. Načte seznam vrstev: `["VodniPlochy", "Silnice_2016", "Lesy"]`
2. Pro každou vrstvu vytvoří buffer 100 metrů
3. Výstupy pojmenuje: `<nazev>_buffer100` (např. `Lesy_buffer100`)
4. Použije funkci `vytvor_buffer(vstup, vzdalenost)`
5. Ošetří chyby pomocí `try-except` (pokud vrstva neexistuje)

8.4 Varianta D (pokročilá)

Vytvořte **Python skript** `multi_buffer.py`, který:

1. Načte libovolnou liniovou vrstvu
2. Stanoví počet obalových zón (`int`) a šířku obalové zóny (`float`)
3. Kolem vstupních linií vytvoří zadaný počet vzájemně navazujících obalových zón, které se nepřekrývají

Poznámka: Nesmíte použít již existující nástroj `Multiple Ring Buffer`!

8.5 Varianta E (výzva pro experty)

Rozšiřte Variantu C:

1. Skript přijímá **parametry z příkazové řádky**:
 - Seznam vrstev (oddělený čárkami)
 - Vzdálenost bufferu
2. Použijte modul `argparse` nebo `sys.argv`
3. Před vytvořením bufferu zkontroluje, zda vrstva existuje (`arcpy.Exists()`)
4. Loguje průběh do souboru `log_buffer.txt`

Příklad spuštění:

```
python buffer_batch.py --vrstvy "rybníky,silnice" --vzdalenost 100
```

9 Cheatsheet

```
# === IMPORT A NASTAVENÍ ===
import arcpy

# Nastavení workspace (geodatabáze)
arcpy.env.workspace = r"C:\Data\Projekt.gdb"

# Nastavení workspace (složka)
arcpy.env.workspace = r"C:\Data\Shapefiles"

# Povolit přepisování výstupů
arcpy.env.overwriteOutput = True

# === INFORMACE O INSTALACI ===
info = arcpy.GetInstallInfo()
print(f"Verze: {info['Version']}")

# === LICENCOVÁNÍ ROZŠÍŘENÍ ===
# Kontrola dostupnosti
status = arcpy.CheckExtension("Spatial")

# Aktivace licence
if status == "Available":
    arcpy.CheckOutExtension("Spatial")
    # ... analýzy ...
    arcpy.CheckInExtension("Spatial")

# === ZÁKLADNÍ NÁSTROJE ===
# Buffer
arcpy.analysis.Buffer(
    in_features="vstup",
    out_feature_class="vystup",
    buffer_distance_or_field="100 Meters"
)

# Clip
arcpy.analysis.Clip(
    in_features="vstup",
    clip_features="hranice",
    out_feature_class="vystup"
```

)

10 Poznámky pro vyučujícího

10.1 Běžné chyby studentů

```
# 1. Zapomínají import arcpy
arcpy.env.workspace = ... # CHYBA - arcpy není definováno
# Správně:
import arcpy
arcpy.env.workspace = ...

# 2. Zapomínají r před cestou
arcpy.env.workspace = "C:\Data\Projekt.gdb" # CHYBA - \D se bere jako escape
# Správně:
arcpy.env.workspace = r"C:\Data\Projekt.gdb"

# 3. Nepovolí overwriteOutput
arcpy.analysis.Buffer("a", "b", "100 Meters") # Chyba pokud "b" existuje
# Správně:
arcpy.env.overwriteOutput = True

# 4. Neaktivují licenci rozšíření
arcpy.sa.Slope("dem", ...) # Chyba - Spatial Analyst není aktivován
# Správně:
arcpy.CheckOutExtension("Spatial")
arcpy.sa.Slope(...)

# 5. Špatná syntaxe nástroje
arcpy.Buffer("vstup", "vystup", 100) # CHYBA - vzdálenost musí být string!
# Správně:
arcpy.analysis.Buffer("vstup", "vystup", "100 Meters")

# 6. Zapomínají .analysis / .management
arcpy.Buffer(...) # CHYBA - Buffer není přímo v arcpy
# Správně:
arcpy.analysis.Buffer(...)
```

10.2 Časový plán (90 min)

Čas	Obsah
0-10 min	Co je ArcPy, kdy použít MB vs. Python
10-30 min	Tři způsoby spuštění (Window, Notebook, Script) - demo každého
30-45 min	První kroky: import, nastavení, licence
45-70 min	Praktický příklad: Buffer třemi způsoby
70-85 min	Praktické cvičení studenty
85-90 min	Shrnutí, výhled na příští lekci, zadání DÚ

10.3 Klíčové momenty

10.3.1 Python Window vs. Notebook vs. Script (10-30 min):

- **UKÁZAT VŠECHNY TŘI** na stejném příkladu (buffer)
- Zdůraznit výhody každého
- Říct: “V praxi budete používat všechny tři - podle situace”

10.3.2 Nastavení workspace (30-45 min):

- **DŮLEŽITÉ:** Ukázat rozdíl mezi workspace = geodatabáze vs. složka
- Zmínit: “Pokud nenastavíte, ArcPy hledá v aktuálním adresáři”
- Ukázat chybu při špatné cestě

10.3.3 Licencování (30-45 min):

- Vysvětlit: “Jen některé nástroje potřebují rozšíření”
- Ukázat: CheckExtension → CheckOutExtension → CheckInExtension
- Říct: “Spatial Analyst - rastrové analýzy, 3D Analyst - 3D, ...”

10.3.4 Buffer - společně (45-70 min):

- Projít VŠECHNY TŘI způsoby na jednom příkladu
- Ukázat “Copy Python Command” z Geoprocessing pane
- Nechat studenty experimentovat

10.4 Rizika

1. Tři způsoby spuštění mohou být matoucí (10-30 min → 35 min)

- Řešení: Důraz na “každý má své místo, není jeden správný”
- Začít Window (nejjednodušší), pak Notebook, nakonec Script

2. Problémy s cestami (r”...”)

- Řešení: Hned na začátku vysvětlit raw string
- Mít připravené příklady různých cest

3. Studenti nemají ArcGIS Pro otevřené

- Řešení: Před lekcí říct “otevřete ArcGIS Pro s nějakým projektem”

10.5 Tipy

- **Mějte připravený projekt** v ArcGIS Pro s ukázkovými daty
- **Sdílejte projekt** se studenty předem (nebo na začátku lekce)
- Na projektoru **ukazujte všechny tři způsoby** vedle sebe
- Používejte **jednoduché příklady** (buffer, clip) - ne složité analýzy
- Zdůrazněte: “Příští týden se naučíte spouštět mimo ArcGIS Pro!”
- **Cheatsheet vytiskněte** a rozdejte studentům

10.6 Materiály k přípravě

- ☐ ArcGIS Pro projekt s ukázkovými daty (silnice, rybníky, lesy)
- ☐ Geodatabáze cviceni_10.gdb ke stažení
- ☐ Příklady kódu pro všechny tři způsoby (Window, Notebook, Script)
- ☐ Screenshots: Python Window, Notebook, Copy Python Command
- ☐ Řešení všech domácích úkolů (A, B, C, D)
- ☐ Diagram: Kdy použít který způsob spuštění