

# Comandos de terminal

Disciplina: Programação aplicada à engenharia cartográfica

Maurício C. M. de Paulo - D.Sc.

26 de fevereiro de 2026

# Tópicos a abordar

- sys.argv: Recebimento de parâmetros
- argparse: Recebimento de parâmetros
- os: manipulação de
- shutil
- pathlib
- glob
- subprocess
- Hydra

# subprocess: Conceitos

**subprocess** permite executar comandos externos a partir do Python.

Principais usos:

- Executar scripts shell
- Chamar programas externos (GDAL, QGIS, etc.)
- Automatizar pipelines

Função mais usada:

- `subprocess.run()`

Boa prática:

- Passar argumentos como lista
- Evitar `shell=True` quando possível

# subprocess: Executando comando com argumentos

```
import subprocess

comando = [
    "gdalinfo",
    "imagem.tif"
]

resultado = subprocess.run(
    comando,
    capture_output=True,
    text=True
)

print("STDOUT:")
print(resultado.stdout)

print("STDERR:")
print(resultado.stderr)
```

## Parâmetros importantes:

- `capture_output=True`
- `text=True` (retorna string)

## subprocess: Exemplo — Salvando stdout e stderr em log

```
import subprocess

comando = ["ls", "-l", "/tmp"]

with open("execucao.log", "w") as log:
    processo = subprocess.run(
        comando,
        stdout=log,
        stderr=log,
        text=True
    )

print("Código de saída:", processo.returncode)
```

### O que acontece:

- stdout → escrito no arquivo
- stderr → escrito no mesmo log
- returncode → indica sucesso (0)