

**SPLN**

# **André Santos**

**MAP-i, FCUP, INESC-TEC**

**[afs@inesctec.pt](mailto:afs@inesctec.pt)**

**[github.com/andrefs](https://github.com/andrefs)**

# **José João Almeida**

**GEPL, DI, UMinho, Algoritmi**

**[jj@di.uminho.pt](mailto:jj@di.uminho.pt)**

**[natura.di.uminho.pt/~jj](http://natura.di.uminho.pt/~jj)**

# **Scripting no Processamento da Linguagem Natural**

**Utilização de linguagens de scripting para o  
desenvolvimento e integração de ferramentas de  
PLN.**

# Conteúdos

- **Introdução ao Python**
- **Scripting**
- **Expressões regulares**
- **Engenharia linguística**
- **Corpora**
- **Gramática do Português**
- **Estratégias de parsing**
- **Semântica**
- **Terminologia**
- **Dicionários e enciclopédias**

# Aulas

- **Python  $\geq 3.6$**
- **Git**
- **Unix**
- **Terminal**
- **Teoria +  
prática**

# Aulas

- [\*\*github.com/andrefs/spln-docs\*\*](https://github.com/andrefs/spln-docs)
  - bibliografia
  - slides
- [\*\*github.com/andrefs/spln-2018-i\*\*](https://github.com/andrefs/spln-2018-i)
  - aulas
  - data
  - usr

**Alunos**

# Avaliação

- **2 ou 3 trabalhos práticos**
- **1 teste escrito**
- **Alguns TPCs**



# Github

- **Criar conta**
  - <https://github.com/join>
- **Pedir Student Pack**
  - <https://education.github.com/pack>
- **Adicionar em**
  - <https://goo.gl/7DnCxC>

# Micro introdução ao Git

- `git init`
- `git add`
- `git`  
`commit`
- `git pull`
- `git push`

# Mini introdução ao Python

```
#!/usr/bin/python3

print("Hello World")

editor = input("What is your favorite text editor? ")

a = [3, 6, 7, 2, 5, 9, 4, 0, 10]

for i in a:
    if i < 5:
        print(str(i) + " is lower than 5")
    elif i > 5:
        print(str(i) + " is higher than 5")
    else:
        print("FIVE!")
```

# Listas

```
list = []
teams = ['slb', 'scp', 'fcp']

teams.append('scb')
teams.extend(['bfc', 'mfcf'])
teams.insert(3, 'vfc')

teams.remove('slb')
teams.pop(3)
del teams[2]
teams[1] = 'lsc'

teams.index('fcp')
teams.count('scp')
```

# Dicionários

```
dict = {}  
colors = {'fcp': 'blue', 'slb': 'red', 'scp': 'green'}  
len(colors)  
'slb' in colors  
colors.get('aca', 'Não existe')  
  
colors.keys()  
colors.values()  
colors.items()
```

# Funções e listas por compreensão

```
import datetime

def timestamp():
    print(datetime.datetime.now())

def repeat(str, n):
    return str * n

words = ['quem', 'como', 'quais']
repetitions = [repeat(word, 2) for word in words]
```

# Ler ficheiros

```
i=0
for line in open('file.txt').readlines():
    i=i+1

print(str(i) + ' lines')
```

# Exercícios

- 1. Programa que pergunta ao utilizador o nome e imprime em maiúsculas.**
- 2. Função que recebe array de números e imprime números pares.**
- 3. Função que recebe nome de ficheiro e imprime linhas do ficheiro em ordem inversa.**
- 4. Função que recebe um texto como argumento e "limpa-o": separa palavras e pontuação com espaços, converte para minúsculas, remove acentuação de caracteres, etc.**
- 5. Função que recebe nome de ficheiro e imprime número de ocorrências das 10 palavras mais frequentes no ficheiro.**



# Referências

## Git

- <http://rogerdudler.github.io/git-guide/>
- <https://services.github.com/on-demand/downloads/github-git-cheat-sheet.pdf>

## Python

- <https://www.practicepython.org/>
- <https://www.python.org/dev/peps/pep-0008/>