

# **Programação II**

**+**

# **Estruturas de Dados para**

# **Bioinformática**

**Apresentação da cadeira**

Hugo Pacheco

DCC/FCUP

22/23

# Programação

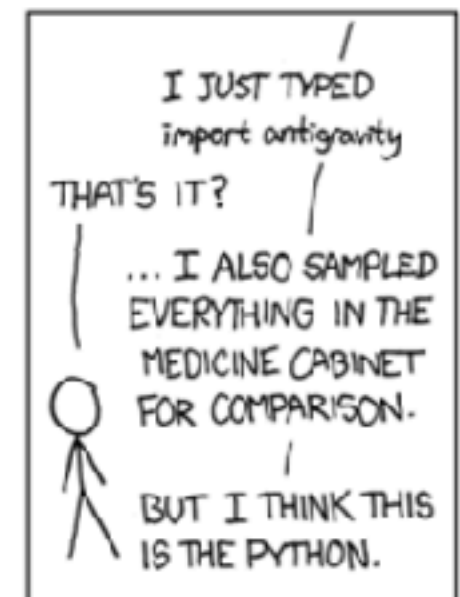
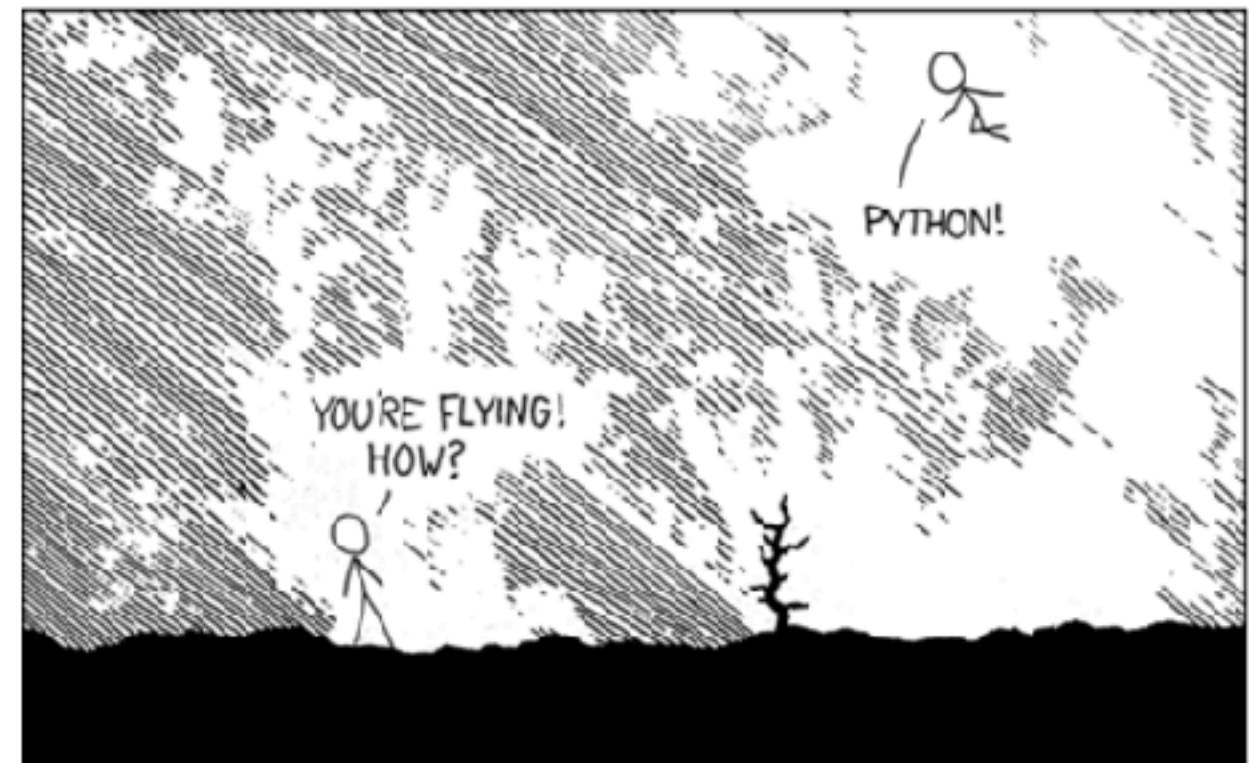
- Programação =
  - Automação de processos
  - Resolução de problemas
  - Extração de informação
  - Visualização de informação

# Programação 101

- Introdução à programação e aos computadores
- Conceitos base de programação em Python
- Apresentação mais **bem-fundada**:
  - Partir dos fundamentos de programação: funções, listas, dicionários, classes, exceções, etc
  - Escrever programas simples que ilustram os conceitos
  - Aprender programação como resolução de problemas

# Python

- Uma linguagem de alto nível e de alta produtividade
- “batteries included”: milhares de bibliotecas prontamente disponíveis
  - gráficos
  - processamento de dados
  - web
- domínio específico (estatística, geografia, biologia, etc)
- ...



# Esta cadeira

- Cadeira profundamente prática de programação em Python
- Apresentação mais **pragmática**:
  - Aprender a utilizar alguns dos melhores recursos de programação para manipulação automática de informação
  - Desenvolver familiaridade com as bibliotecas e ambiente de desenvolvimento existentes e melhorar experiência de programação
  - O objetivo não é formar especialistas em Ciências da Computação. Tanto quanto possível, não vamos nem queremos olhar para dentro do “capô”

# Análise de dados

- É uma fração significativa de toda a programação que é feita hoje em dia
- É a principal forma de programação feita por programadores não profissionais
- Python é uma linguagem bastante forte neste domínio
- Pode ser aplicada em domínios variados, não necessariamente por especialistas de informática: biologia, ciências sociais, medicina, geografia, economia, etc
- “Data-driven world”: grandes quantidades de dados, necessidade de automatização e extração de conhecimento

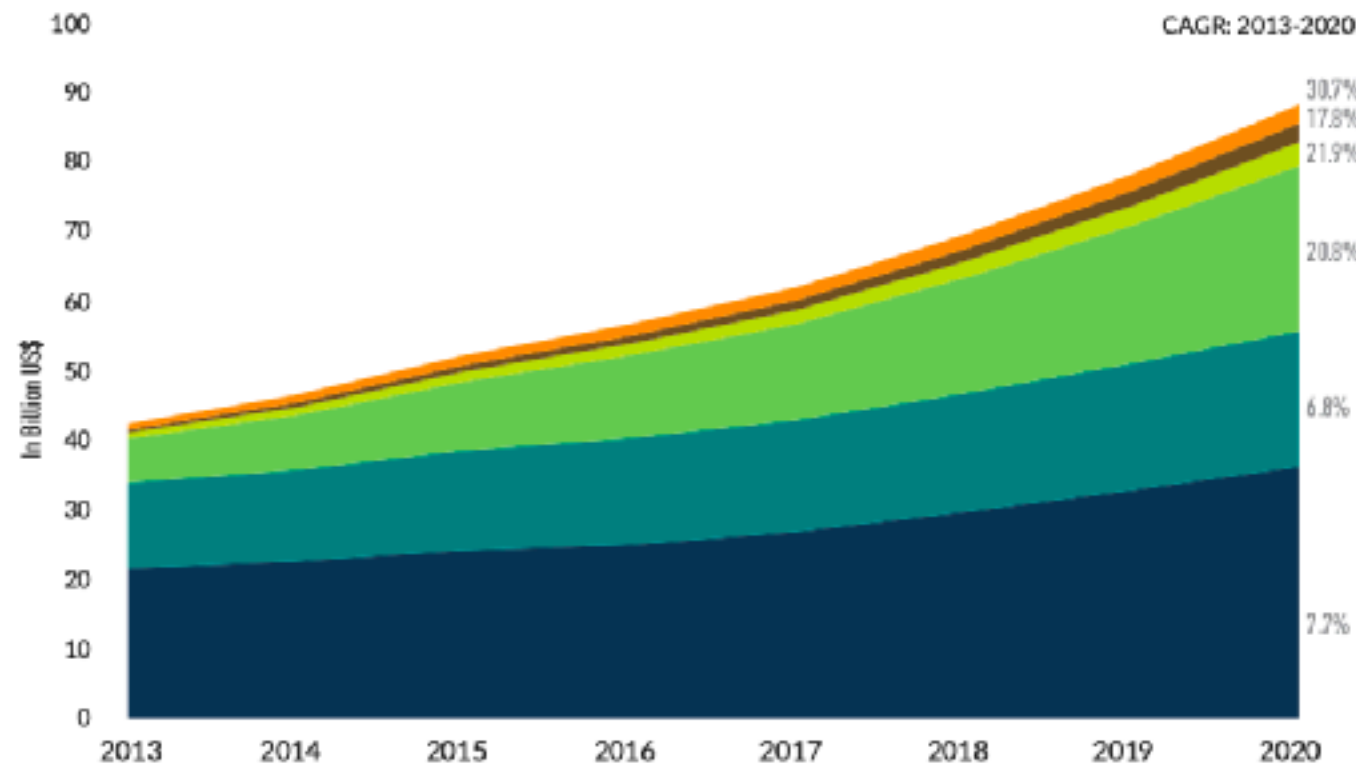
90% of the world's data  
was generated over the last  
2 years...

[source from 2020]



# Análise de dados

- Geoespacial

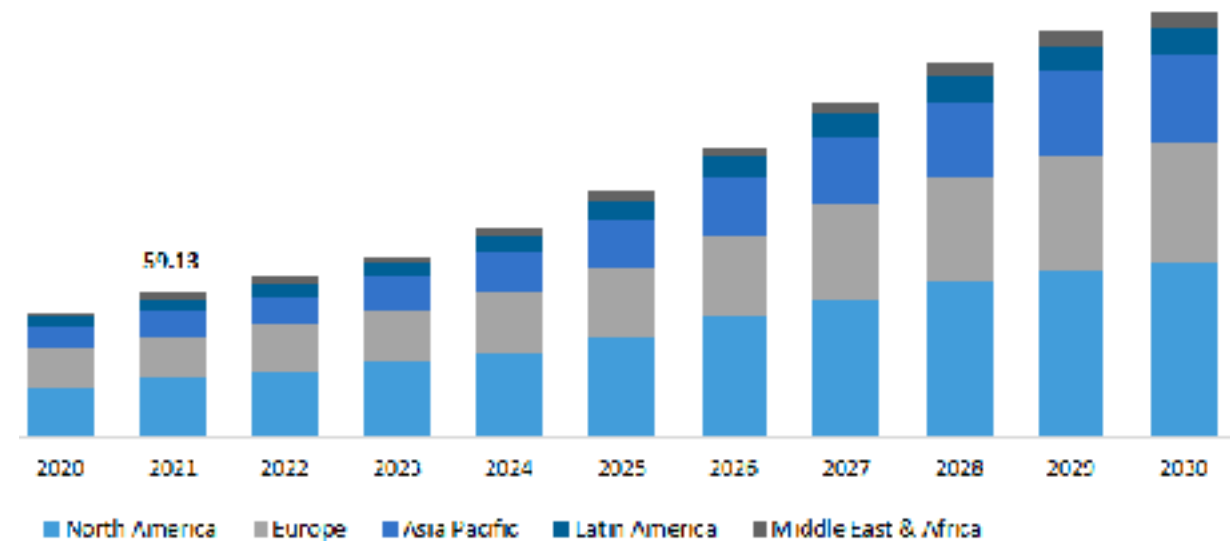


Market Share (2020)

Region	Market Share (%)
North America	41.1%
Europe	22.0%
Asia Pacific	26.9%
South America	3.8%
Middle East	3.0%
Africa	3.1%

[source for Geospatial market growth]

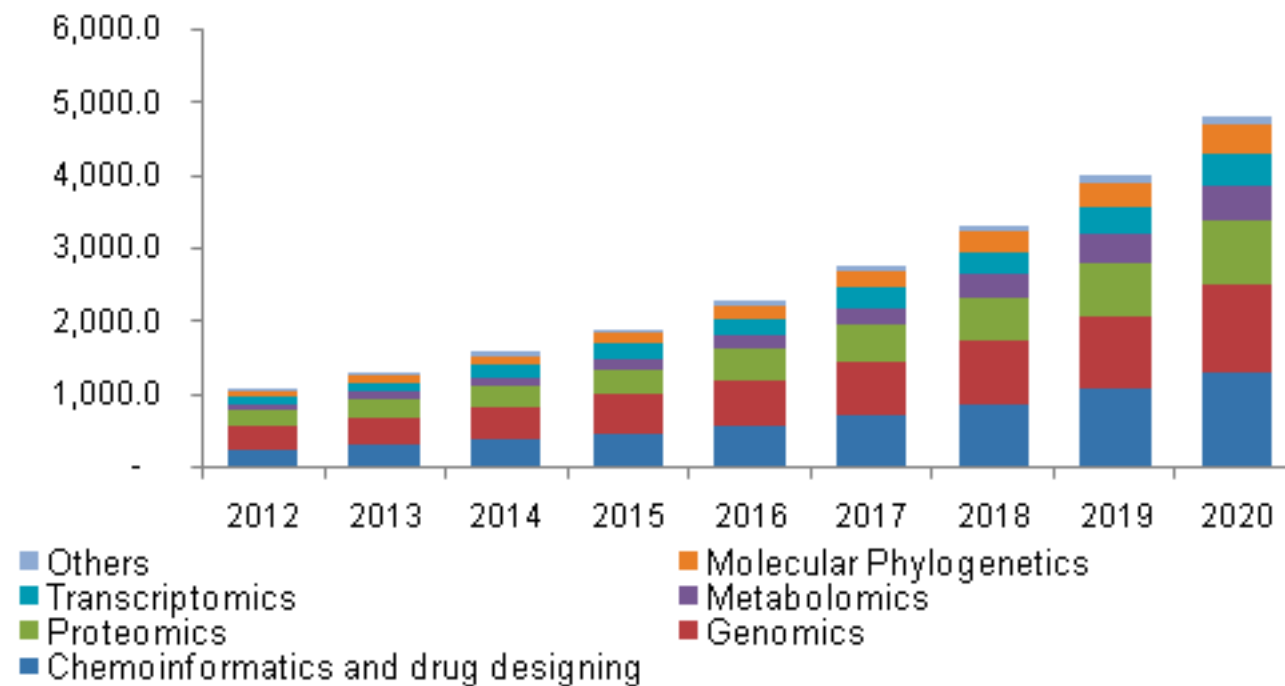
Geospatial Analytics Market Size, By Region, 2018 - 2030  
(USD Billion)



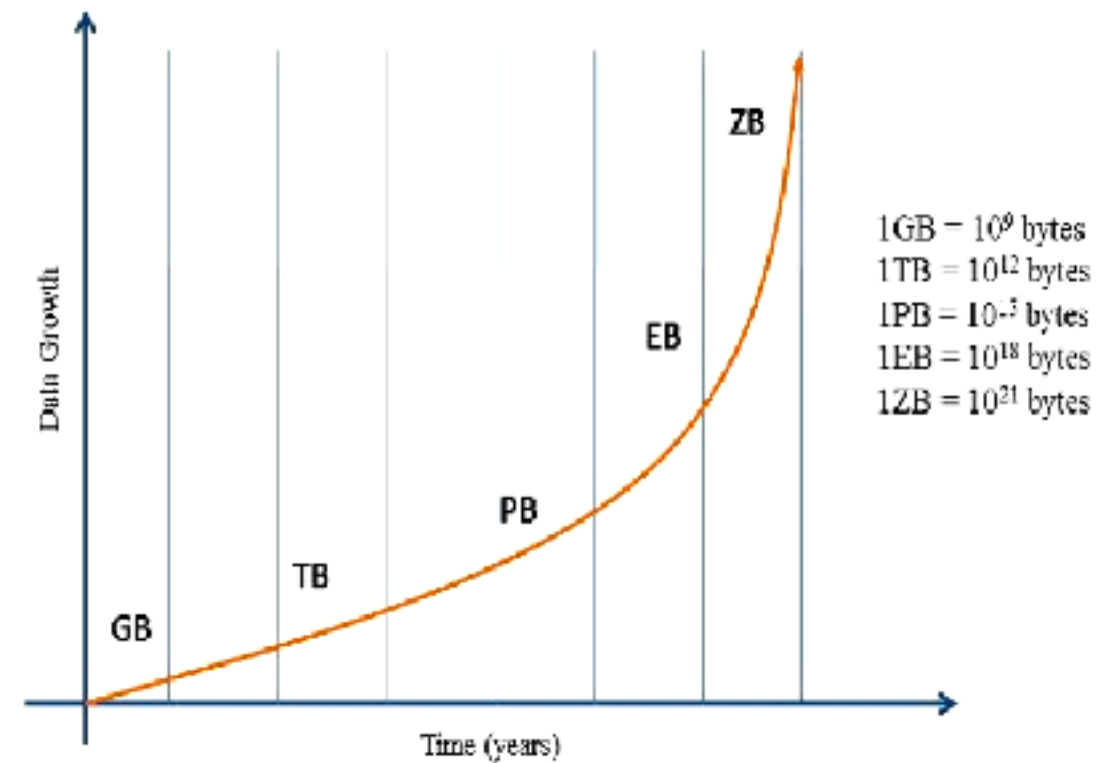
[source for predicted Geospatial market growth]

# Análise de dados

- Bioinformática



[source for the European Bioinformatics Market from 2020]



[source for Bioinformatics data size from 2018]



# The Rise of Data Scientists

BEFORE



nobody cared for a  
"math geek" in parties.

NOW



People love ~~math~~geeks!  
data scientists!  
RK

WHEN A USER TAKES A PHOTO,  
THE APP SHOULD CHECK WHETHER  
THEY'RE IN A NATIONAL PARK...

SURE, EASY GIS LOOKUP.  
GIMME A FEW HOURS.

... AND CHECK WHETHER  
THE PHOTO IS OF A BIRD.

I'LL NEED A RESEARCH  
TEAM AND FIVE YEARS.



IN CS, IT CAN BE HARD TO EXPLAIN  
THE DIFFERENCE BETWEEN THE EASY  
AND THE VIRTUALLY IMPOSSIBLE.



www.biocomicals.com

# Programa

## 1. Revisão dos conceitos base

- números, strings, listas, dicionários, classes, etc

## 2. Análise de dados

- formatos comuns de partilha de informação
- modelos comuns de representação e manipulação de dados

## 3. Visualização de dados

- diferentes formas de visualização de dados (gráficos, mapas)
- Construção de animações para visualização de dados

# Avaliação

- Não serão registadas presenças (teóricas e práticas)
- Avaliação completamente prática
- **70%** nota prática:
  - 3 trabalhos práticos **individuais** para serem desenvolvidos nas aulas práticas e em casa
- **30%** nota oral: defesa individual do trabalho no fim do semestre
  - elegância, organização e documentação do código
  - capacidade de resposta a perguntas sobre o trabalho
  - componente contínua (acompanhei o vosso trabalho nas aulas práticas)
- É possível melhorar e entregar os trabalhos em época de recurso, mas será exigida nova funcionalidade; têm que ter frequência (ter entregue os trabalhos) em época normal

# Trabalhos práticos

- O enunciado de cada trabalho estará dividido em Tarefas.
- Podem utilizar os enunciados de anos anteriores como exercícios adicionais para treinar
- Cotações de cada Tarefa serão decididas mais à frente consoante o progresso das aulas
- Algumas Tarefas, quando indicado, poderão servir como valorização (a decidir consoante o progresso das aulas)
- Esta não tenciona ser uma cadeira técnica: os principais critérios de avaliação serão **esforço** e **interesse**

# Trabalhos práticos

## 1. Análise de texto

- processamento do texto de *A Relíquia* de Eça de Queiroz: contar palavras, agrupa, etc
- análise simples de sequências de DNA: detectar enzimas

## 2. Análise de dados (**anos anteriores, TBD**)

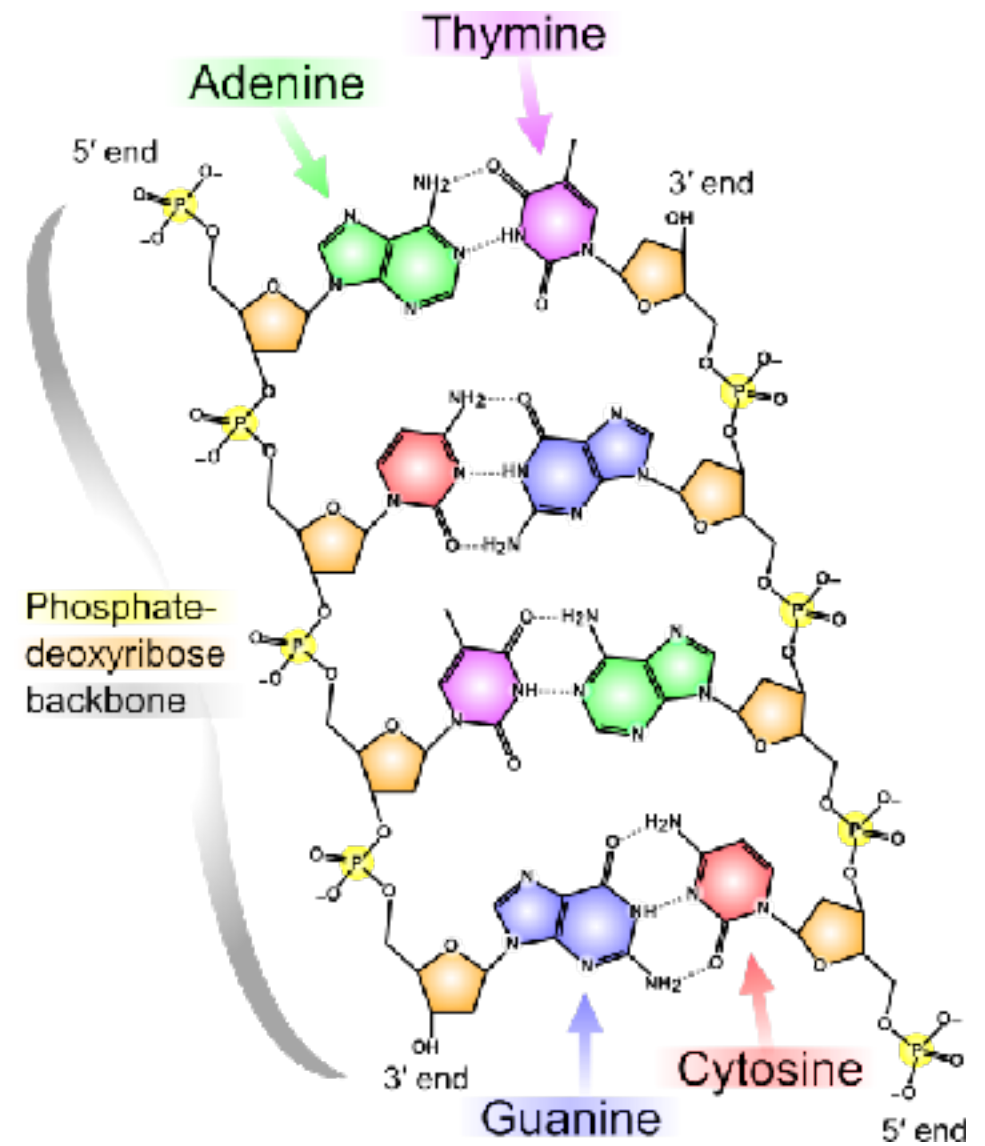
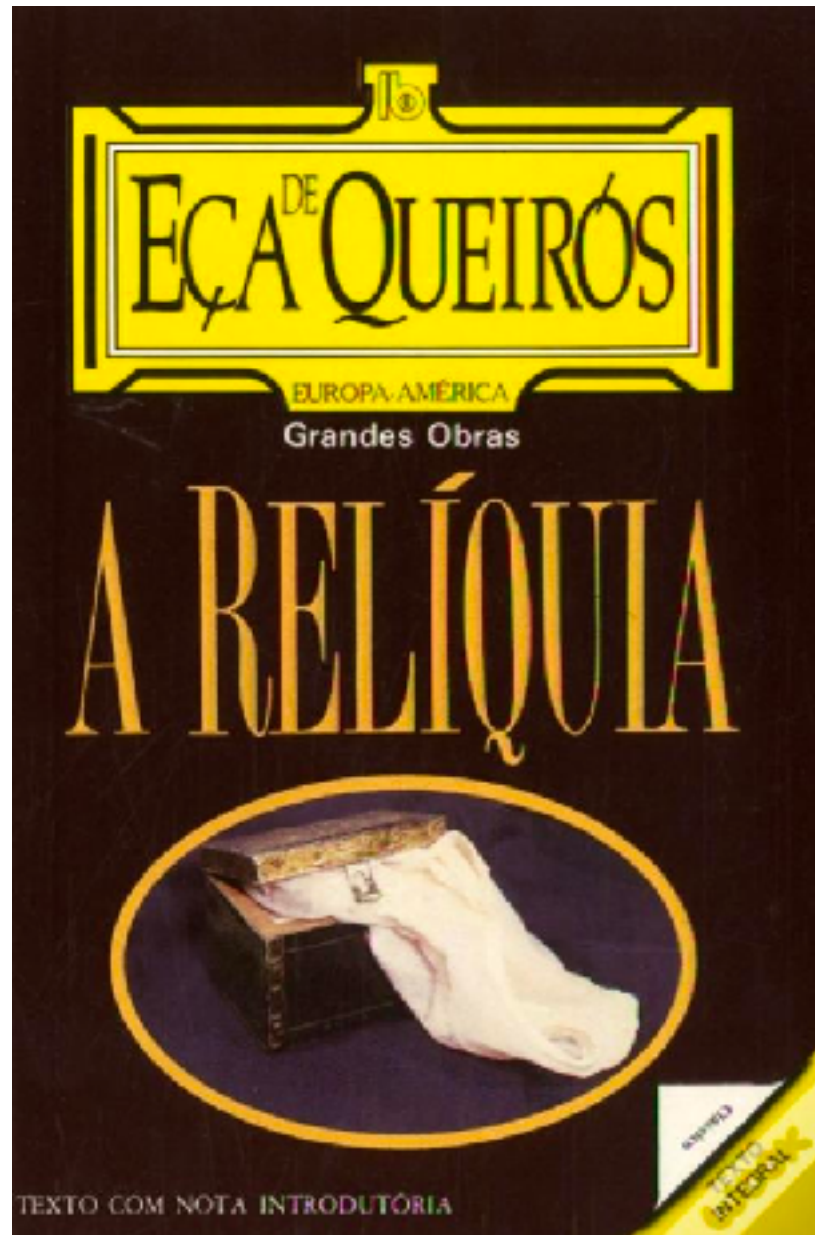
- análise de dados fornecidos pelo IPMA: previsão meteorológica, sismicidade, etc
- análise de dados de consumo de energia: nuclear, renováveis, etc

## 3. Visualização e animação de dados (**anos anteriores, TBD**)

- análise e visualização de dados sobre a pandemia COVID-19 publicados pela DGS
- Análise e visualização de dados sobre o conflito na Ucrânia: refugiados, ataques, etc

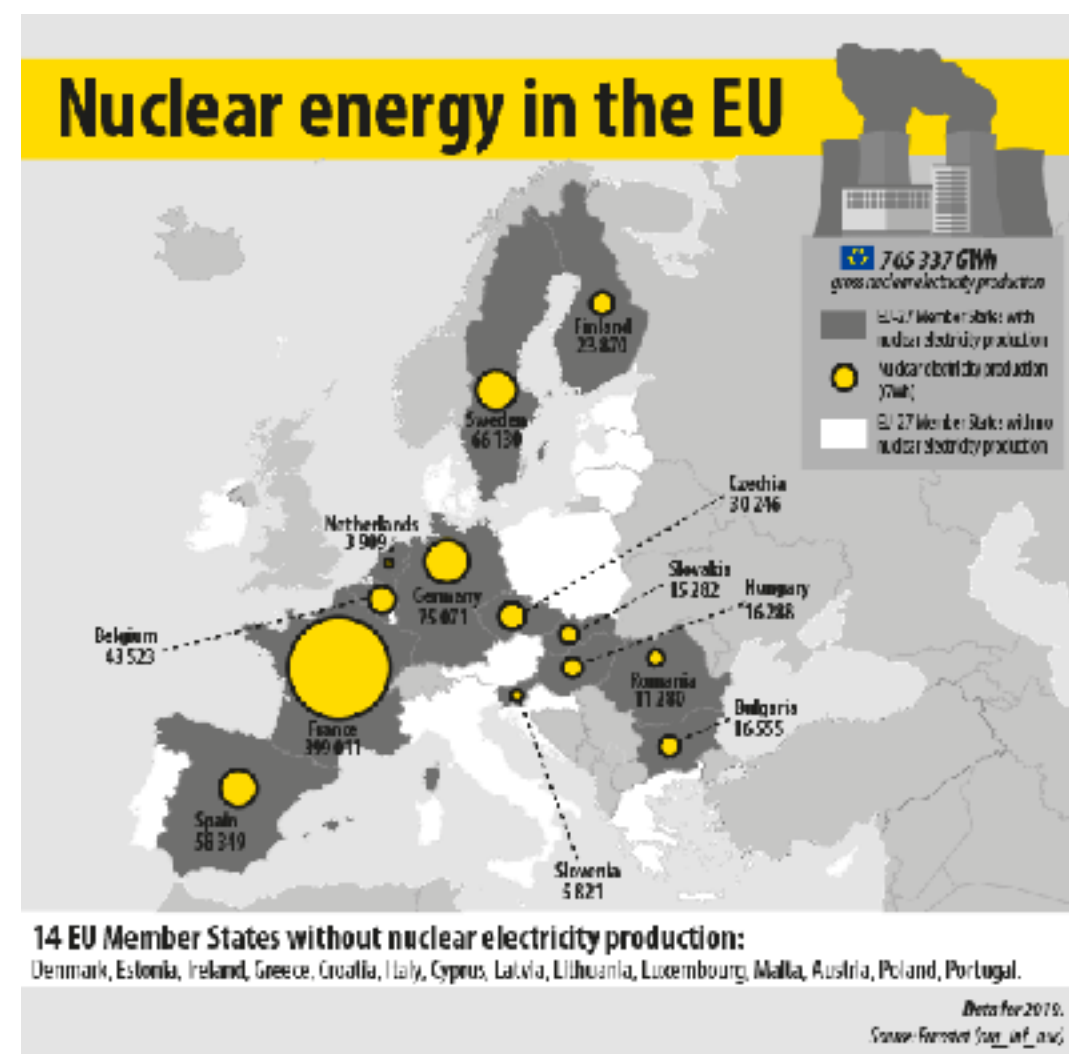
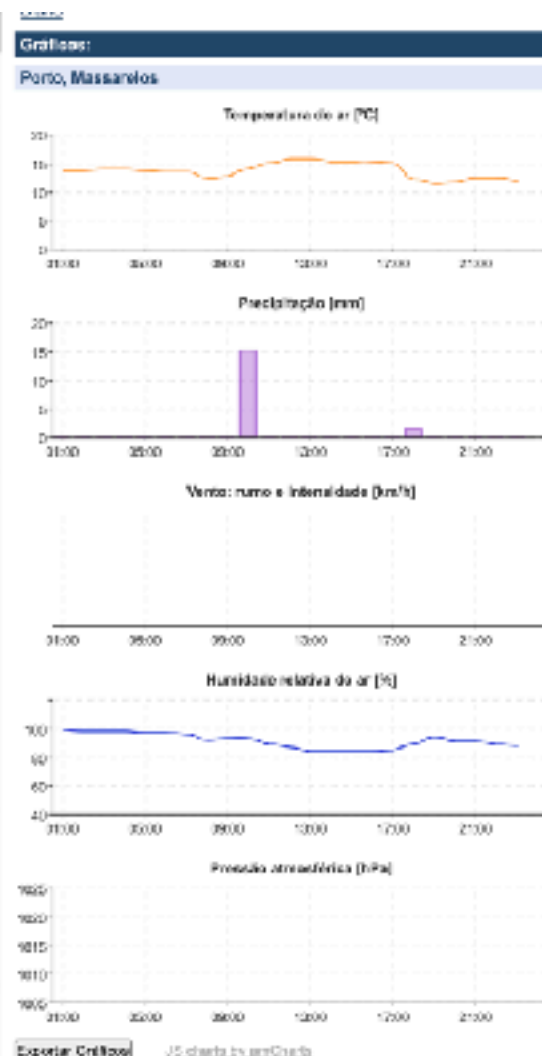
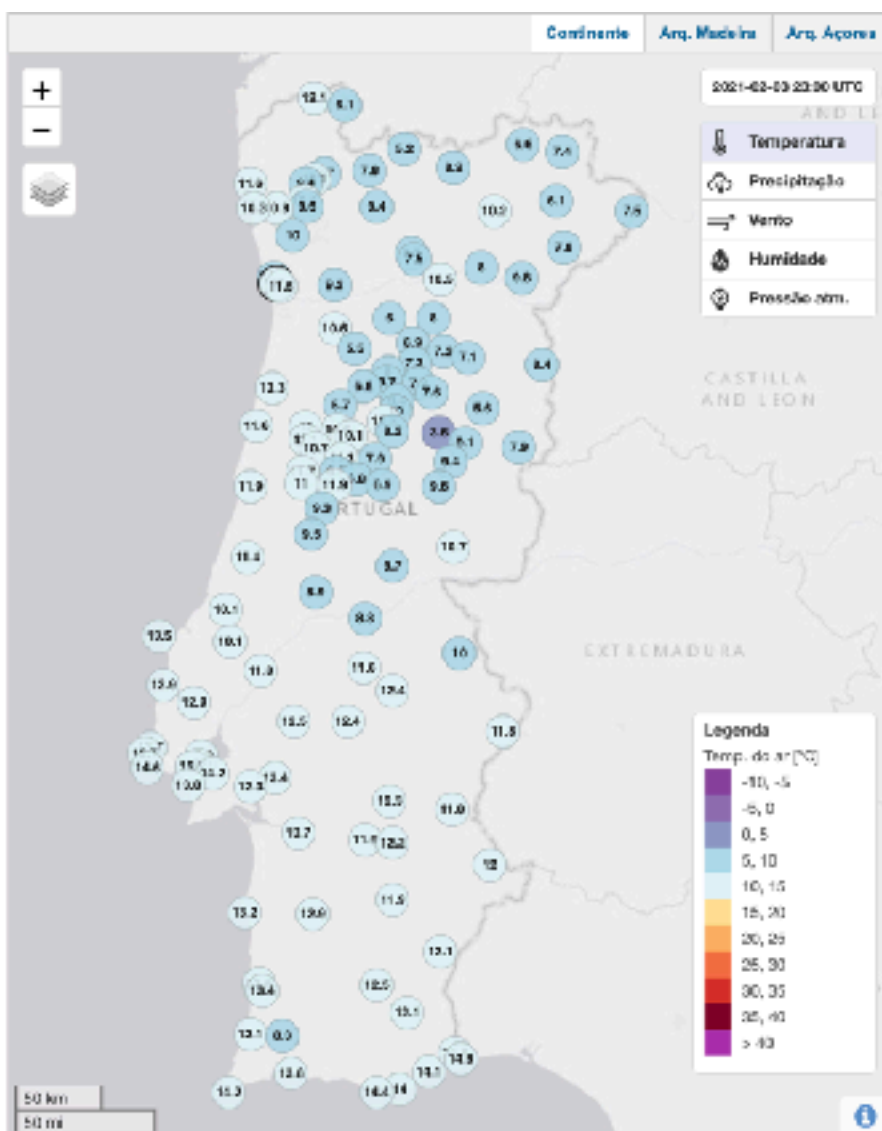
**TBD:** Temas ainda por decidir

# Projeto 1

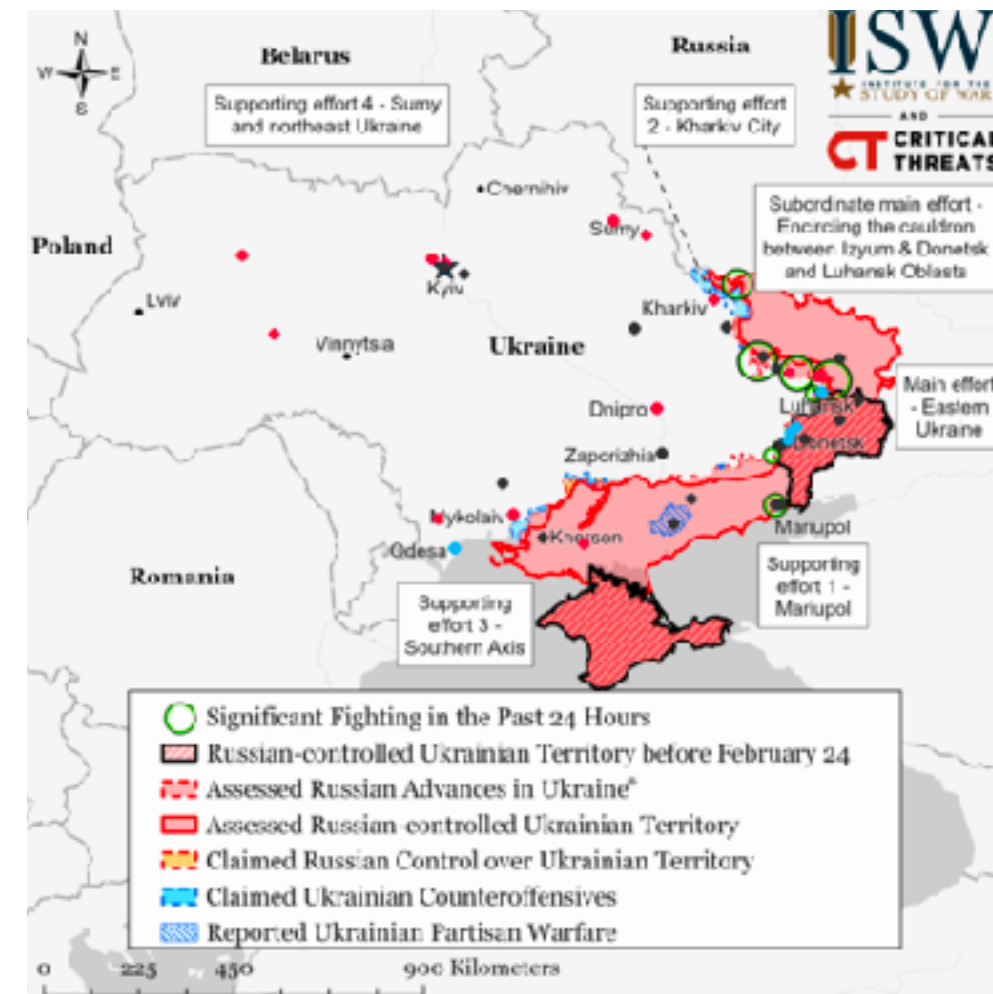
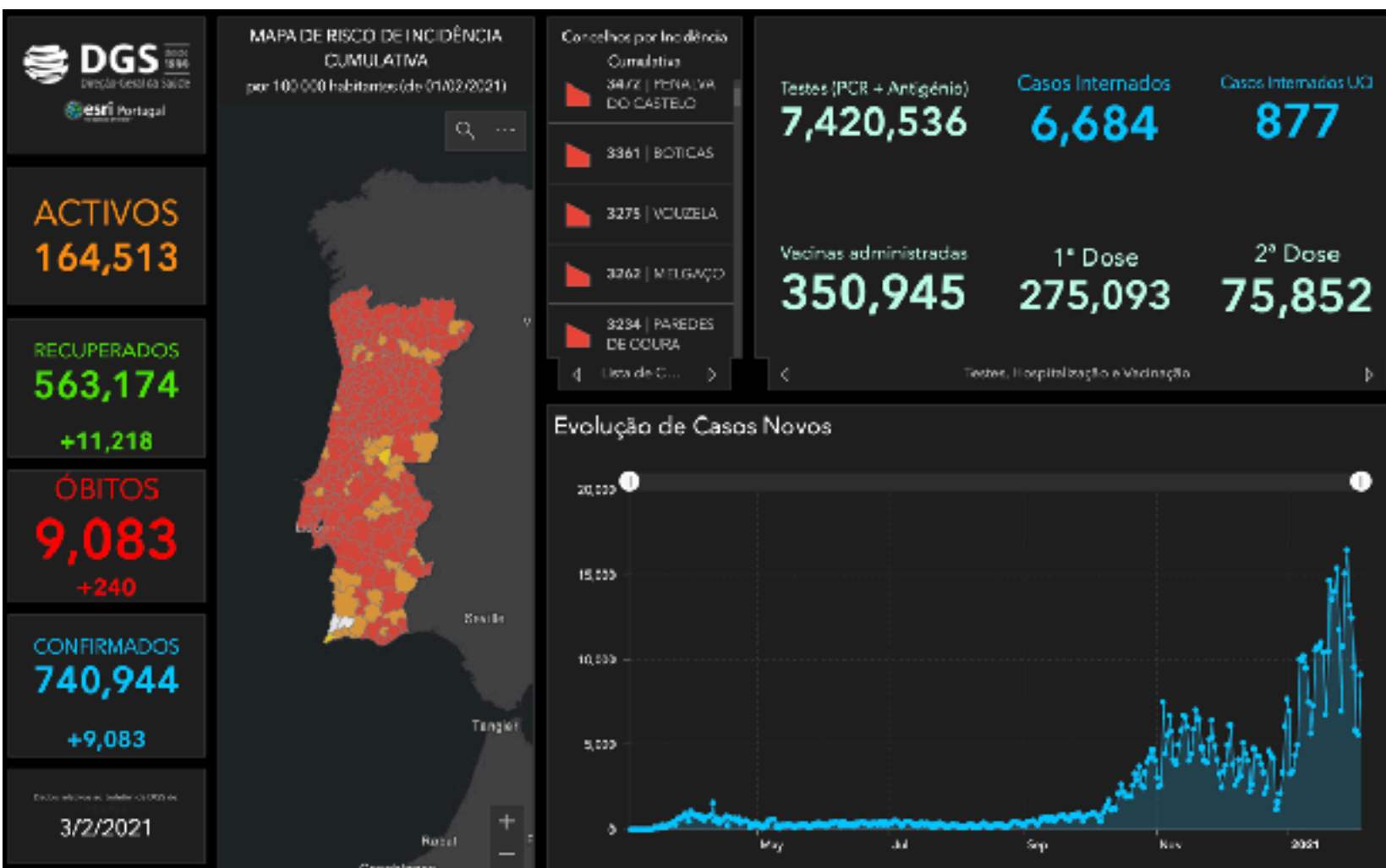




# Projeto 2



# Projeto 3





# Entregas

- Entrega por email para o docente ([hpacheco@fc.up.pt](mailto:hpacheco@fc.up.pt))
  - Enviar apenas um ficheiro de código com o nome projeto?.py
  - Não enviar ficheiros de suporte (texto, json, imagens, gráficos, ...)
  - Não alterar o template (assinaturas de funções e código pré-definido)
  - (Alternativa) submeter o projeto no [replit](https://replit.com)
- Datas **provisórias**, podem vir a ser alteradas com o calendário
  - Projeto 1: entrega até 19 Março
  - Projeto 2: entrega até 30 Abril
  - Projeto 3: entrega até 01 Junho

# Informações Gerais

- Sigarra: site oficial da cadeira
- Github: repositório com todo o material de suporte à cadeira (slides, projetos práticos, etc)
- MS Teams: plataforma para comunicação fora das aulas; instant messaging

# Aulas

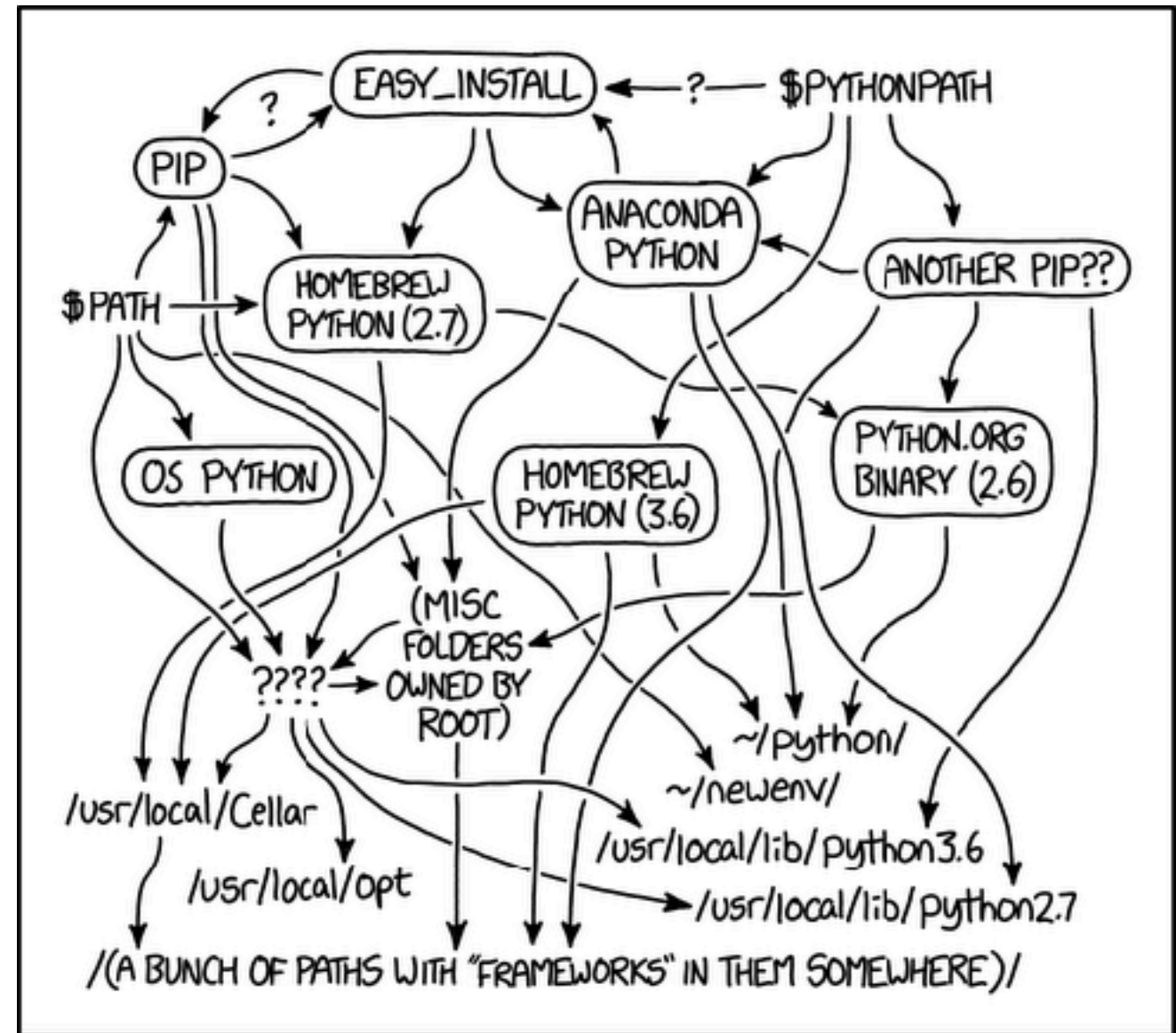
- Em regime **presencial**
- Aulas teóricas
  - apresentação da matéria e exemplos práticos de utilização
- Aulas práticas
  - resolução dos guiões dos projetos práticos

# Docentes

- Hugo Pacheco ([hpacheco@fc.up.pt](mailto:hpacheco@fc.up.pt)):
- Não existe horário de atendimento pelo menos por enquanto. Apela-se à comunicação pelo **MS Teams** ou por email institucional

# TPC: Instalar o Python

- Existem muitas formas de instalar o Python e bibliotecas associadas: *binário, anaconda, pip, brew, apt-get*, etc
- Se já estão familiarizados com um ambiente, usem esse; evitem ter múltiplas instalações do Python



# TPC: Instalar o Python

- Como sugestão...
- Instalar Python 3.x “nativo”
  - Windows: fazer download do instalador oficial em <https://www.python.org/downloads/>
  - Linux: *sudo apt-get install python3*
  - Mac OS: *brew install python3*
- Utilizar o editor PyCharm CE (<https://www.jetbrains.com/pycharm/download/>)
  - code highlighting, debugging, documentação, etc
  - gestão e instalação automática de pacotes, não interfere com outras possíveis instalações

# Projeto 0 - Desenferrujar

- Primeiras duas semanas de aulas práticas para rever matéria e aferir conhecimentos
- Enunciado do Projeto 0
- Antes da aula:
  - Instalem o Python
  - (Alternativa) editor online replit, login com conta Google UP (upXXXXX@g.uporto.pt)