

# Programação I

## Aula 1 — Apresentação

Pedro Vasconcelos  
DCC/FCUP

## Nesta aula

1 Objetivos, funcionamento e avaliação

2 Introdução aos computadores

# Conteúdo e objetivos

- Noções sobre computadores e pensamento computacional
- Introdução programação usando a linguagem Python
- No fim deste curso deverá ser capaz de:
  - 1 simular a execução de programas simples passo-a-passo;
  - 2 aprender algumas estruturas de dados e algoritmos fundamentais;
  - 3 codificar algoritmos simples em Python;
  - 4 decompor problemas em subrotinas e funções re-utilizáveis;
  - 5 testar e corrigir erros dos seus programas.

## Funcionamento

**Teóricas** (anfiteatro) exposição de conceitos; exemplificação com problemas de programação  
2 × 1 h por semana

**Práticas** (laboratórios) consolidação de conhecimentos; resolução de exercícios supervisionada  
2 h por semana

NB: reserve ainda cerca de 2h por semana para estudo e resolução de exercícios fora das aulas.

# Aulas teóricas

O que fazer / não fazer...

- ✓ Chegar a horas
- ✗ Entrar ou sair durante durante a aula
- ✓ Tomar notas (mas os *slides* vão estar disponíveis)
- ✓ Colocar questões (ordeiramente)
- ✗ Conversar para o lado (demasiado barulho)

# Aulas laboratoriais

O que fazer / não fazer...

- ✓ Chegar a horas
- ✓ Assinar a folha de presenças
- ✗ Entrar ou sair durante durante a aula
- ✓ Tentar resolver exercícios propostos
- ✓ Rever as notas e *slides* das aulas teóricas
- ✓ Pedir ajuda (docente e/ou colegas)
- ✗ Esperar que o docente resolva os exercícios
- ✗ Copiar resoluções sem compreender

# Avaliação

- Teste intermédio (20%)
- Exame final (80%)

Condições para obtenção de frequência:

- 1 Presença em 2/3 (dois terços) das aulas práticas
- 2 Resolução de 1/2 (metade) dos exercícios de aulas práticas com correção automática

**NB: se peder frequência **não poderá fazer exame em época normal nem de recurso!****

## Bibliografia recomendada

*Think Python* (2ª edição) A. B. Downey.

<http://greenteapress.com/wp/think-python-2e/>.

Livro de distribuição gratuita.

# Bibliografia complementar

(Apenas para quem quiser continuar para tópicos mais avançados.)

*Python: How to Program*. H. M. Deitel, P. J. Deitel, J. P. Liperi, B. A. Wiedermann.

Disponível na biblioteca.

## Outros recursos

**Página web** slides de aulas, folhas de exercícios

`http:`

`//www.dcc.fc.up.pt/~pbv/aulas/programacaoI`

**Página de submissão de exercícios** para avaliação automática

`https://codex.dcc.fc.up.pt/cc1015`

**Fórum de discussão** perguntas sobre aulas, exercícios e submissões

`https://piazza.com/fc.up.pt/spring2019/cc1015/home`

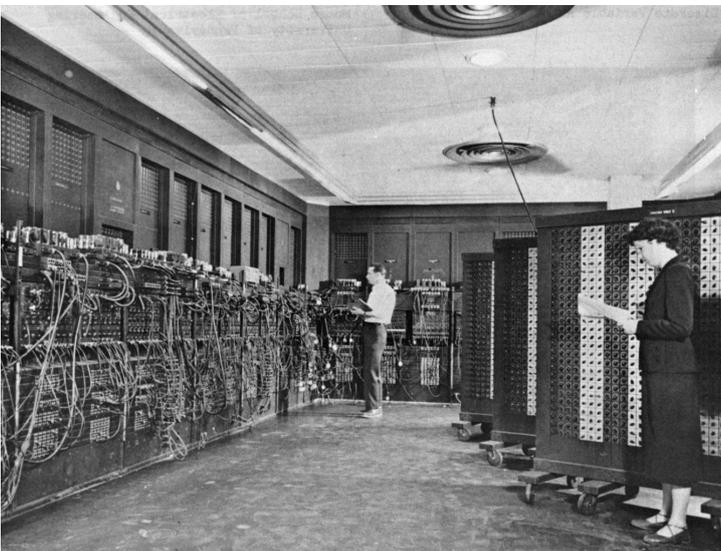
# Introdução aos computadores

## Computador

- um processador de **informação** (dados);
  - segundo um **programa** (lista de instruções).
- 
- Efectua rapidamente muitas operações aritméticas:
    - humano**: aproximadamente 1 operação/segundo
    - computador actual**: >1 milhão de operações/segundo
  - Processa **informação matematizável** (ex: números, textos, sons, imagens...)
  - Utilizável para diversos fins com diferentes **programas**

## Breve cronologia dos computadores

1940s **Computadores eletro-mecânicos e a válvulas**: Colossus, Harvard Mk I, ENIAC



ENIAC, U.S. Army photo

## Breve cronologia dos computadores (cont.)

1950s **Primeiros computadores comerciais**: UNIVAC I  
1960s **“Mainframes”**: IBM System/360



IBM System/360 exposto no *Deutsches Museum in Munich Germany*

## Breve cronologia dos computadores (cont.)

1970s **Mini-computadores**: DEC VAX, PDP-11, UNIX



PDP-11, 1972

# Breve cronologia dos computadores (cont.)

1977–1982: **Primeiros computadores pessoais**



Apple II



IBM PC



Sinclair ZX Spectrum



Acorn BBC Micro

# Breve cronologia dos computadores (cont.)

1984–1986: **Computadores pessoais com interfaces gráficas**



Apple Macintosh



Commodore Amiga



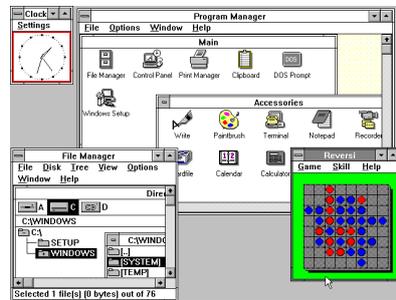
Atari ST

# Breve cronologia dos computadores (cont.)

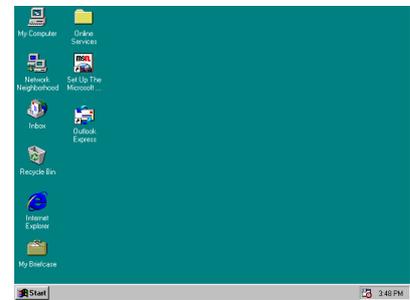
1985–1990: “IBM PC clones” e a arquitetura “Wintel”



Windows 1.0



Windows 3.0



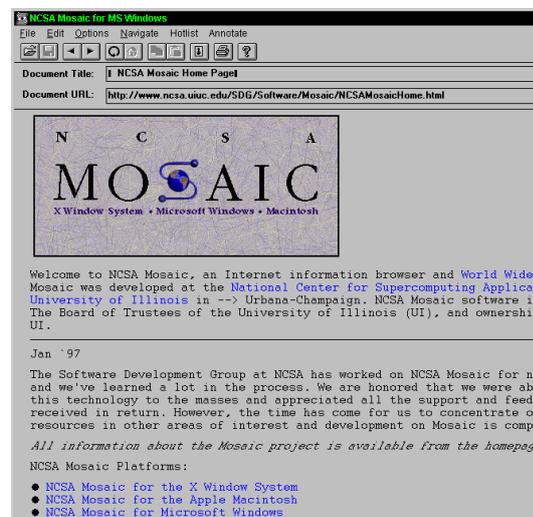
Windows 95

# Breve cronologia dos computadores (cont.)

1991: World Wide Web



Primeiro servidor web (CERN)



NCSA Mosaic 1.0 web browser

# Breve cronologia dos computadores (cont.)

1991: GNU/Linux



Richard Stallman (fundador da *Free Software Foundation* e do projeto *GNU*)



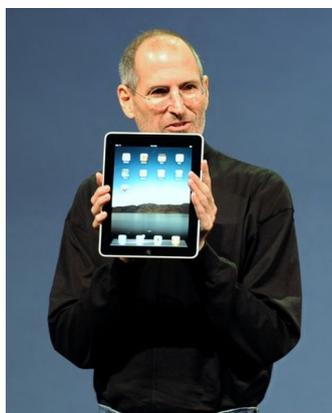
Linus Torvalds (criador do *kernel Linux*)

# Breve cronologia dos computadores (cont.)

2007–2008: Plataformas móveis



Apple iPhone



Apple iPad



Google Android Nexus

# Níveis conceptuais de um computador

**Hardware:** CPU, memória, unidades de disco, ecrã, teclado, rato. . .

**Software:** sistema operativo, aplicações, jogos, ficheiros (imagens, músicas, filmes, folhas de cálculo, bases de dados. . .)

## Níveis conceptuais de um computador (cont.)

Tendências:

- *hardware* mais barato
- *software* mais complexo e caro
- importância do desenvolvimento de *software*:
  - utilização de linguagens de alto nível
  - re-utilização de componentes (bibliotecas)

# Sistemas de operação

- Conjunto de *software* de base para gerir recursos do computador
- Proporciona **funcionalidades** para as aplicações:
  - gestão de utilizadores
  - gestão de memória
  - gestão de ficheiros
  - gestão de input/output (I/O): terminais, impressoras, interfaces gráficas, ligações de rede

## Cronologia do sistemas de operação

Primeiros sistemas de operação (1950s):

- um trabalho de cada vez (*batch*)
- apenas supervisiona as transições entre trabalhos

*Mainframes* (1960-1970s)

- caros: necessário partilhar recursos
- multi-utilizador: vários utilizadores em *terminais*
- multi-tarefa: divisão o tempo de processamento entre os vários trabalhos (*time-sharing*)

# Cronologia do sistemas de operação (cont.)

## Mini-computadores: sistema UNIX (1970s)

- multi-utilizador, multi-tarefa
- portátil para diferentes modelos de computadores
- código-fonte em linguagem C distribuído com o sistema
- popular na comunidade académica (Universidade de Berkeley)
- variantes comerciais: Ultrix, System V, IRIX, Solaris

# Cronologia do sistemas de operação (cont.)

## Primeira geração de computadores pessoais (1970–1982)

- um utilizador, uma tarefa
- interface textual
- muitas vezes combinada com um interpretador de BASIC
- específico para cada modelo de computador

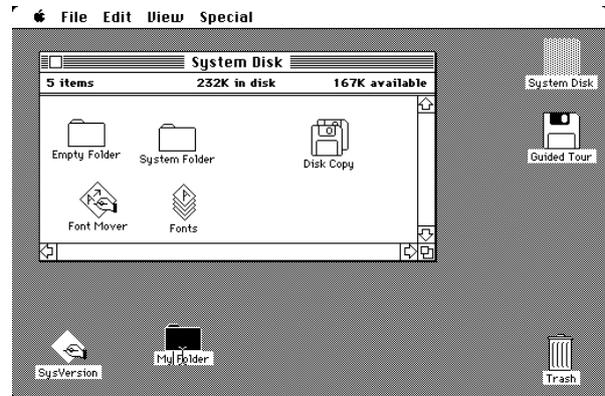


```
HELLO, WORLD!  
LIST  
10 HOME  
20 INVERSE  
30 PRINT "HELLO, WORLD!"  
40 NORMAL  
50 PRINT CHR$(?)
```

# Cronologia do sistemas de operação (cont.)

## Primeiras interfaces gráficas (1984–1990)

- um utilizador, multi-tarefa
- aplicações associadas a janelas independentes
- apresentação de programas e ficheiros usando ícones
- seleção usando um “rato”
- específico para cada modelo de computador



# Cronologia do sistemas de operação (cont.)

## Actualmente:

- computadores pessoais com mais recursos do que os antigos super-computadores
- ligados em **redes locais e globais** (internet)
- sistemas multi-utilizador, multi-tarefa
- maior independência do *hardware*
- separação entre o **núcleo** e a **interface gráfica**
- UNIX em PCs: GNU/Linux, MacOS X