

PROGRAMAÇÃO E ALGORITMOS (LEI, TSI)

Universidade da Beira Interior, Departamento de Informática
Hugo Pedro Proença, 2016/2017

Página da Disciplina

<http://www.di.ubi.pt/~hugomcp/progAlg>

- Avaliação.
 - Consulta das classificações atribuídas a cada item de avaliação.
- Acetatos teóricos e fichas práticas.
- Anúncios.
 - Informação relevante sobre a disciplina.
 - Datas limite para entrega de trabalhos, novidades, etc...
- Links.
 - Tutoriais, manuais, e outros recursos disponíveis on-line e de interesse para a disciplina.
- Etc...

Pré-Requisitos I

Pré-Requisitos:

Tópicos que se assumem assimilados previamente pelos alunos. Apesar de **não incluídos no conteúdo** programático da disciplina, o sucesso nesta **exige o conhecimento** de cada um dos itens apresentados à direita.

□ Sintaxe da Linguagem C

- ▣ Estrutura de um Programa
- ▣ Palavras Reservadas
- ▣ Blocos Condicionais
- ▣ Blocos Iterativos

Pré-Requisitos II

Pré-Requisitos:

Tópicos que se assumem assimilados previamente pelos alunos. Apesar de não incluídos no conteúdo programático da disciplina, o sucesso nesta exige o conhecimento de cada um dos itens apresentados à direita.

- **Edição, Compilação e Execução de Programas em C**
 - ▣ Escolha de Editor de Texto
 - ▣ Bibliotecas
 - Bibliotecas mais Comuns
 - Inclusão de Bibliotecas
 - *Linguagem* de Bibliotecas

Objectivos



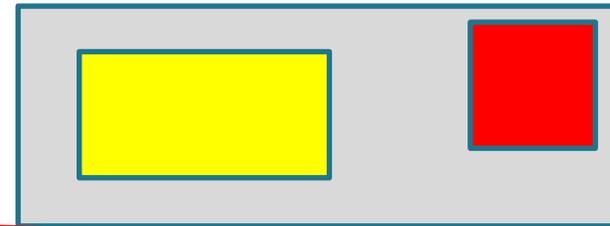
- ❑ O objectivo principal desta disciplina é a compreensão de conceitos fundamentais relacionados com o custo computacional de um processo automático:
 - ❑ Alocação de Recursos
 - ❑ Recursividade
 - ❑ Ordenação
 - ❑ Pesquisa
- ❑ A linguagem C será usada na componente prática da disciplina, apesar dos conceitos nela envolvidos serem independentes da linguagem.

Objectivos: Estruturas Dinâmicas?

- Em termos gerais, existem três formas de alocar recursos (memória) num computador:

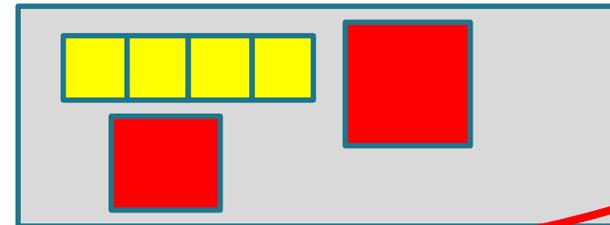
- **Estaticamente**

- Tempo de compilação.



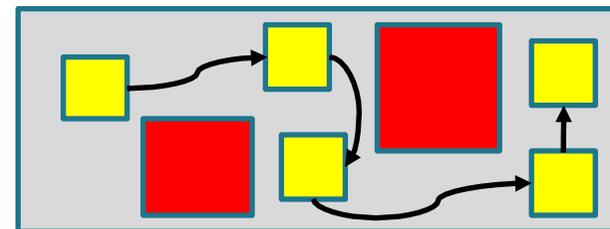
- **Vectores**

- Tempo de execução.
- Memória adjacente.
- Acesso directo.



- **Estruturas ligadas.**

- Tempo de execução.
- Memória descontínua.
- Acesso Sequencial.



Critérios de Avaliação

Assiduidade (A)

- Serão registadas as presenças nas aulas teóricas e práticas. 15 semanas letivas, 15 aulas teóricas + 15 aulas práticas. Cada falta não-justificada (Serviços Académicos) implicará uma penalização de 0.25 valores na nota final (frequência e exames).
- Consideram-se dispensados do critério de assiduidade os trabalhadores/estudantes registados com tal estatuto nos Serviços Académicos da Universidade.

Frequências

- Prova (F1) - 2ª feira, 24 de abril de 2017, 11:00-13:00, Sala 6.03 (7 valores)
- Prova (F2) - 3ª feira, 03 de junho de 2017, 11:00-13:00, Sala 6.03 (7 valores)

Trabalho Prático

- O trabalho prático da disciplina é individual.
- (6 valores) Media Searching**

- [\[MEDIA.zip\]](#)

Classificação Ensino/Aprendizagem (C)

- A classificação relativa ao período de Ensino Aprendizagem é obtida através da seguinte fórmula:
 - $C = F1 * 10 / 20 + F2 * 10 / 20 + T - A$

Admissão a Exame

- Consideram-se admitidos a Exame os alunos que obtiverem classificação mínima de 6 valores na componente de Ensino-Aprendizagem.

Exames

- Provas E1, E2 e Especial (Data, hora e sala a definir pelos Serviços Académicos).

Programa



- Introdução
 - ▣ Input / Output: Streams e Ficheiros Associados
- Alocação de Memória
 - ▣ Memória Estática
 - ▣ Memória Dinâmica
 - Sequencial
 - Disjunta
- Pré-processador
 - ▣ Directivas
 - ▣ Macros
- Complexidade Computacional
- Ordenação
- Pesquisa

Bibliografia Recomendada

□ Principal

- "Introduction to Algorithms (2nd edition)", Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, The MIT Press, 2001.
- "C Programming Language", Brian Kernighan and Dennis Ritchie, Prentice Hall, ISBN: 978-0131103627

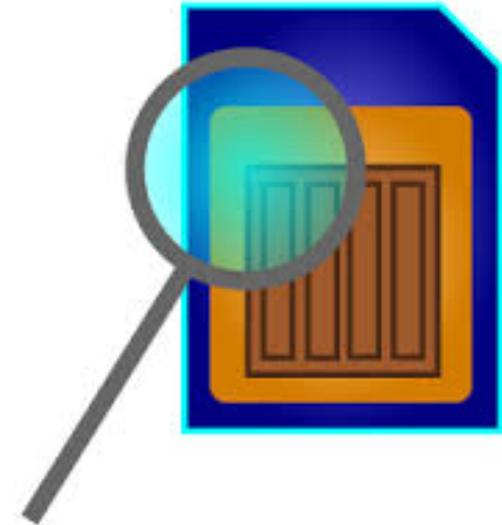
□ Secundária

- "Mastering algorithms in C", Kyle Loudon, O'Reilly, 1999.
- "A Book on C: Programming in C", Al Kelley and Ira Pohl, Addison-Wesley Professional, 978-0201183993.

Trabalho Prático: Media Searching

- Os conteúdos multimédia (audio, video), representam hoje uma parte significativa do volume de dados da Web
- Suponha (em termos abstratos) que se pretende implementar um sistema de pesquisa de conteúdo multimédia, que opera ao nível do bit:
- Exemplo:

0,1,1,0,0,1,,0,0,1,1,0,1,0,0,1,0,1,1,0...



Trabalho Prático: **Media Searching**

□ **Trabalho**

- Suponha que uma polícia forense Internacional possui uma base de dados de conteúdos multimédia.
- Para cada conteúdo, guarda-se um identificador e a sequência multimédia (binária)
- O objectivo é que para uma dada pesquisa binária, se devolvam os IDs de multimédia mais parecidos com a sequência introduzida

Trabalho Prático: **Media Searching**

- O trabalho, a implementar em linguagem ANSI C, consiste:
 - ▣ **na leitura do ficheiro de texto** com a informação dos Ids e conteúdos multimédia;
 - ▣ **carregamento em memória;**
 - ▣ **pesquisa** dos elementos mais parecidos a um excerto de um conteúdo;
 - ▣ Devolução dos “*top-n matches*”
 - Top-1: o mais parecido
 - Top-2: os dois mais parecidos
 - ...

Trabalho Prático: *Media Searching*

□ Exemplo de funcionamento:

MENU

1:Leitura de ficheiro

2:Pesquisa de conteúdo

Trabalho Prático: Media Searching

□ Notas:

- Os segmentos de multimédia têm comprimento fixo (N=2048).
- Os segmentos a procurar podem ter dimensão variável (K, $1 < K < 2048$).
- Exemplo. Procurar por “0110010”.
 - Deverão ser devolvidos os conteúdos que contenham sequências mais próximas da pedida em qualquer parte da sua assinatura:
 - ID1: 01000**0110010**000101010... (Distância 0)
 - ID2: **0110010**001011100101010 (Distância 0)
 - ID3: **1110010**001011100101010 (Distância 1)

Trabalho Prático: *Media Searching*

□ Exemplo do Ficheiro *MEDIA.txt*

```
12345678 \t 100100110...  
11004562 \t 0101000110...
```

...