

Teoria da Computação
2022-2023
(cod.14343/14813)

Departamento de Informática
Universidade da Beira Interior

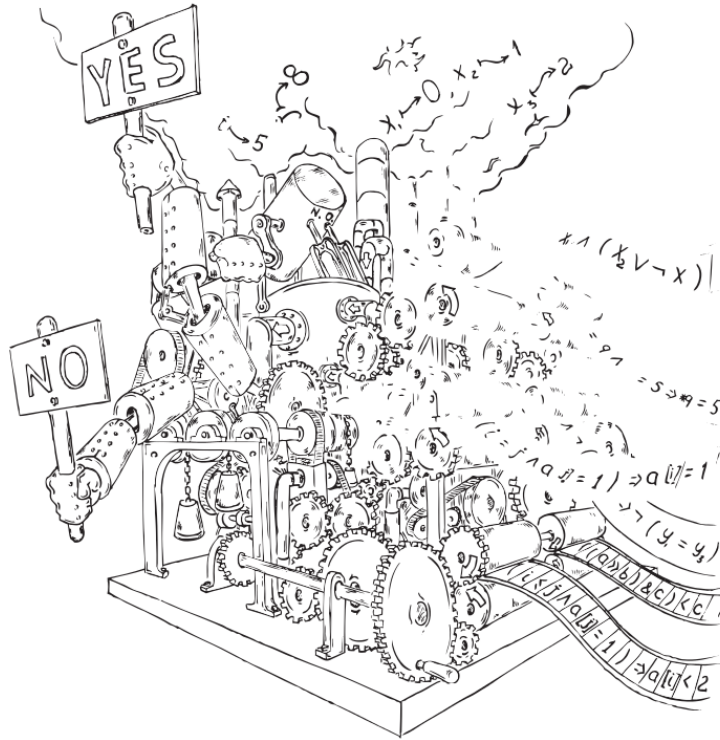


Figura 1: Decision Procedure

Esta página no formato pdf
Paper do David Hilbert sobre o seu programa para a matemática do século
XX (“os 23 problemas”) aqui
Paper de Alan Turing introduzindo as Máquinas de Turing aqui
Paper de Alonzo Church introduzindo o Cálculo Lambda aqui
Paper de Lance Fortnow sobre o estatuto recente do problema $P \stackrel{?}{=} NP$
aqui

1 Novidades

- Usaremos o Teams **22-23_(Agregação) TEORIA DA COMPUTAÇÃO** para a gestão contínua da UC de Teoria da Computação.
- Regra de Ouro:

Qualquer tipo de fraude em qualquer dos itens de avaliação implica a reprovação imediata.

- O sistema mooshak será utilizado para a avaliação prática na presente disciplina. Queira proceder ao seu registo. Aceita-se a formação de grupos de, no máximo, duas pessoas. No processo de registo, escolhe o grupo UBI e defina o nome da equipa da seguinte forma.

Se for um grupo de um só elemento: número de aluno + primeiro nome. Por exemplo, Luís com o número 12345 tem por registo mooshak "12345Luis".

Se for um grupo de duas pessoas: número de aluno do primeiro elemento (o de número mais baixo) + primeiro nome+número de aluno do segundo elemento + primeiro nome. Assim se Luís forma grupo com o João (aluno 13245), então o grupo regista-se com o nome "12345Luis13245Joao".

- Como colocar uma dúvida ao regente da Unidade Curricular?
 1. Comparecer nas aulas e colocá-la directamente ao regente
 2. Usar o **Teams** da Unidade Curricular
 3. Comparecer no horário de atendimento do regente e colocá-la directamente

4. enviar um email ao regente (desousaUUU@UUUdi.ubi.pt, (retire os UUU)) com o assunto "TC: XXXX"em que XXX é o título da dúvida em questão. Qualquer outro formato no assunto arrisca condenar o email ao esquecimento.
- Inscrição em turmas práticas: via site dos serviços académicos.
 - As aulas práticas começam logo na primeira semana de aulas.
 - Os alunos com estatuto de trabalhador estudante devem procurar o regente para discutir eventual adaptação dos critérios de avaliação.
 - Primeira versão da página. Encontrará aqui as novidades associadas à disciplina de **Teoria da Computação**. A sua consulta regular é necessária ao bom funcionamento da Unidade Curricular.

Conteúdo

1	Novidades	2
2	Docentes	4
3	Objectivos	4
4	Programa	4
4.1	Competências da UC ou Resultados da Aprendizagem	5
5	Critérios de Avaliação	5
5.1	Actividades de Ensino-Aprendizagem e Metodologias Pedagógicas	5
5.2	Fraudes	6
5.3	Componente Prática	6
5.4	Componente Teórica	6
5.5	Concessão de Frequência e Avaliação Contínua	6
5.6	Avaliação por Exame	7
6	Datas Importantes	7
7	Horário	8

8	Atendimento	8
9	Material Pedagógico e Funcionamento da Disciplina	8
10	Bibliografia Principal	10

2 Docentes

- Simão Melo de Sousa (regente, T e PL) - Gabinete 3.17 - Laboratório Release (6.25) - Bloco VI.
- Luis Horta (PL).

3 Objectivos

Existem limites à capacidade de resolução de problemas por um computador, mesmo na hipótese “idealista” de ausência de restrições, que sejam essas o tempo (de execução) ou o espaço (memória).

Para delinear esses limites, visaremos:

1. perceber a capacidade de computação das máquinas, assim como os seus limites teóricos. Precisaremos de definir formalmente o que é e o que não é um programa, um algoritmo, ou mais genericamente o que é um tratamento efectivo;
2. perceber os conceitos que fundamentam as linguagens de programação. Precisaremos de determinar e estudar formalmente as construções que determinam a expressividade (ou capacidade de computação) das linguagens de programação assim como o comportamento dos programas.

4 Programa

- Apresentação Contextual e Histórica da Teoria da computação
- Modelos da computação: dos autómatos (de estados finitos, com pilha) às máquinas de Turing.

- Modelos de computação alternativos: Funções recursivas de Kleene e calculo lambda.
- Programação em modelos da computação.
- Tese de Church-Turing. Provas de equivalência de modelos.
- A não computabilidade e a indecidibilidade: Problemas indecidíveis, técnica da diagonalização, técnica da redução.
- Complexidade. Introdução: problemas tratáveis e problemas intratáveis. Critérios de catalogação (memória, tempo, etc.). Caracterização das classes NP, P e NP-Completo. Problemas NP-Completo: Exemplos. Técnica da redução polinomial para a demonstração de NP-completude.

4.1 Competências da UC ou Resultados da Aprendizagem

O aluno deverá ser capaz de perceber e usar a capacidade de computação das máquinas, assim como os seus limites teóricos.

Deverá ser capaz de formalizar adequadamente e avaliar se determinados problemas tem solução computacional ou não.

Deverá perceber e saber usar modelos, técnicas e algoritmos de computação simbólica introduzidos na resolução de problemas informáticos do dia-a-dia.

5 Critérios de Avaliação

5.1 Actividades de Ensino-Aprendizagem e Metodologias Pedagógicas

Por fim a avaliar as competências adquiridas, as actividades de Ensino-Aprendizagem avaliarão tanto a compreensão dos conceitos teóricos expostos como a capacidade em por estes em prática.

Assim, a avaliação será constituída por duas componentes: a componente **prática** (exercícios práticos entregues à equipa docente) e a componente **teórica** (provas escritas).

Mais precisamente a avaliação será realizada por uma prova escrita e por avaliação contínua baseada na resolução de exercícios práticos.

5.2 Fraudes

A equipa docente realça que qualquer tipo de fraude em qualquer dos itens desta disciplina implica a reprovação automática do aluno faltoso (i.e. Não Admissão), podendo ainda vir a ser este alvo de processo disciplinar.

Listamos a seguir as diferentes componentes da avaliação.

5.3 Componente Prática

- Esta avaliação mede em termos práticos a aquisição dos conceitos expostos. Como tal é baseada na avaliação da resolução de exercícios durante as práticas laboratoriais.
- Os **2** exercícios avaliados são resolvidos de uma forma sequencial. Os enunciados serão publicados no início do semestre. As datas exactas de entrega encontram-se na secção 6. A entrega é feita de forma electrónica no site mooshak da UC.
- A Nota da Componente Prática (NCP, 20 valores) é a soma dos valores atribuídos aos diferentes exercícios resolvidos.

5.4 Componente Teórica

A avaliação da componente teórica consiste numa única prova escrita (frequência), prevista na data anunciada na secção 6.

Da avaliação qualitativa da frequência resulta a Nota da Componente Teórica (NCT, 20 valores).

5.5 Concessão de Frequência e Avaliação Contínua

O parâmetro de "Frequência" atribuído no final desta unidade curricular traduz, no contexto da avaliação contínua, a "avaliação mínima" do estudante ao longo do processo de ensino-aprendizagem no final das actividades de contacto.

Considera-se que o estudante demonstrou ter adquirido o grau de conhecimentos mínimos (durante o processo de aprendizagem ao longo das actividades lectivas) quando este demonstrou as mínimas competências em cada componente avaliada.

É assim concedido **Frequência** ao aluno que entregou com sucesso (“accepted” no mooshak) pelo menos um dos exercícios da componente de avaliação prática.

No caso de Frequência, a avaliação quantitativa, designada aqui de nota da avaliação contínua, é determinada da seguinte forma:

$$\frac{\text{componente prática (NCP)} \times 0.8 + \text{componente teórica (NCT)} \times 1.2}{2}$$

Se a avaliação quantitativa resultar numa nota maior ou igual a 10 então o aluno é dispensado de exame (Frequência com dispensa de exame).

5.6 Avaliação por Exame

- A prova escrita do exame substituirá a Nota da Componente Teórica da avaliação contínua, dando uma nova NCT.
- Assim a nota final (NFin) após exame é calculada da seguinte forma:

$$NFin = \text{if } (NCT \geq 6) \text{ then } \frac{NCT \times 1.2 + NCP \times 0.8}{2} \text{ else } \textit{Reprovado}$$

6 Datas Importantes

- Data da Frequência : 3 de Janeiro de 2023, das 18h00 às 20h00 (sala por anunciar).
- Entrega do primeiro exercício : até a semana do 28 de Novembro de 2022.
- Entrega do segundo exercício : semana do dia 13 de Janeiro de 2022.
- Exames : (conferir no site dos académicos)

7 Horário

Tipo de aula	Horário	Sala	Docente
Teórica	Segunda-Feira das 11h00 às 13h00	6.01	S. Melo de Sousa
Práticas Laboratoriais	Terça-Feira das 9h00 às 11h00	6.13	S. Melo de Sousa
Práticas Laboratoriais	Quarta-Feira das 9h00 às 11h00	6.13	Luís Horta
Práticas Laboratoriais	Quarta-Feira das 11h00 às 13h00	6.13	Luís Horta

(TBD: Por definir)

8 Atendimento

Por marcação (e.g. via email) ou

Horário	Docente
Segunda-Feira das 14h00 às 16h00	S. Melo de Sousa

9 Material Pedagógico e Funcionamento da Disciplina

Os Apontamentos serão atempadamente disponibilizados nas aulas e por meios electrónicos. É esperado e assumido que o aluno tenha lido os acetatos referentes ao capítulo em curso antes das aulas teóricas.

Teóricas

Computação Simbólica e Programação consultar a página seguinte (link)

Modelos de Computação

- Aula Inaugural - O nascer de uma disciplina - Introdução contextual e histórica à Teoria da Computação (em forma de sebenta)
- Capítulo: complementos sobre técnicas matemáticas

- Complementos sobre técnicas matemáticas para a Teoria da Computação (mais detalhes sobre indução estrutural) - suporte pedagógico auxiliar
- Capítulo: Conceitos preliminares à Teoria da Computação e à Teoria das Linguagens Formais
- Capítulo: Autómatos Finitos
 - Acetatos manuscritos "automatos" num só pdf
 - Acetatos manuscritos "automatos" formato RAR
- Capítulo: Álgebras de Kleene e expressões regulares
- Capítulo: Algoritmia das strings
- Capítulo: Linguagens e Gramáticas
- Capítulo: Autómatos de Pilha e Linguagens Algébricas
- Capítulo: Bibliotecas e Ferramentas para o processamento de linguagens
- Capítulo: Máquinas de Turing
- Capítulo: Funções Recursivas de Kleene

Computabilidade e Teoria da Complexidade

- Capítulo: A não-computabilidade
- Capítulo: A teoria da complexidade

Código OCaml para suporte às aulas

- Execução genérica de autómatos finitos não determinísticos com ϵ -transições (em ocaml)
- Execução genérica de autómatos finitos determinísticos (em ocaml)

Práticas

- Ficha: Fundamentos da Computação
- Ficha: Técnicas Matemáticas
- Ficha: Introdução às Linguagens Formais
- Ficha: Linguagens Regulares e Autômatos
- Ficha: Linguagens regulares e os seus limites
- Ficha: Gramáticas, Linguagens Formais e Linguagens Algébricas
- Ficha: Máquinas de Turing
- Ficha: Funções recursivas de Kleene
- Ficha: A não computação
- Ficha: Complexidade algorítmica

Algumas resoluções

- Zip contendo enunciados de provas de anos anteriores
- Correção da Frequência 2011/2012

Exercícios Por Entregar

Por Definir.

- Problema A: Por definir
- Problema B: Por definir

10 Bibliografia Principal

As referencias principais são: [10, 7, 8, 12, 1] utilizaremos ocasionalmente as referências [2, 3, 11, 9, 5, 4, 6].

Referências

- [1] J.B. Almeida, M.J. Frade, J.S. Pinto, and S. Melo de Sousa. Rigorous Software Development, An Introduction to Program Verification, volume 103 of Undergraduate Topics in Computer Science. Springer-Verlag, first edition, 307 p. 52 illus. edition, 2011.
- [2] A. Arnold and I. Guessarian. Mathematics for Computer Science. Prentice-Hall, 1996.
- [3] Olivier Carton. Langages formels, Calculabilité et Complexité. Vuibert Ed., June 2014.
- [4] E. Chailloux, P. Manoury, and B. Pagano. Developing applications with objective caml. <http://caml.inria.fr/oreilly-book>, 2003.
- [5] Chris Hankin. Lambda Calculi: A Guide for Computer Scientists, volume 3 of Graduate Texts in Computer Science. Clarendon Press, Oxford, 1994.
- [6] Jason Hickey, Anil Madhavapeddy, and Yaron Minsky. Real World OCaml. O'Reilly, 2014.
- [7] J.E. Hopcroft, R. Motwani, and J.D. Ullman. Introduction to automata theory, languages, and computation. Pearson education, third edition, 560 pages. edition, 2006.
- [8] Dexter Kozen. Automata and Computability. Springer-Verlag, New York, 1997.
- [9] Dexter Kozen. Theory of Computation. Springer, New York, 2006.
- [10] Harry R. Lewis and Christos H. Papadimitriou. Elements of the Theory of Computation. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1997.
- [11] P. Linz. An introduction to formal languages and automata. Jones and Bartlett Publisher, 2006.
- [12] M. Sipser. Introducton to the Theory of Computation. PWS Publishing, 2006.

Enviar comentários e dúvidas para (retire os UUU) : desousaUUU@UUUdi.ubi.pt