

LICENCIATURA EM CIÊNCIAS DA COMPUTAÇÃO
(1º CICLO)

Proposta de Adequação

Janeiro 2006

Dossier Interno

Conteúdo

1	Enquadramento e Justificação	1
2	Objectivos	2
3	Resultados Esperados da Aprendizagem	2
4	Perfil de Formação	3
5	Estrutura do Curso e Plano de Estudos	3
6	Recursos Humanos e Materiais	5
7	Saídas Profissionais	5
8	Encargos	7
9	Calendarização, Processo de Transição e Equivalências	7
A	Minuta de Resolução do Senado Universitário	11
B	Ficha relativa ao enquadramento no Conselho de Cursos	14
C	Proposta de Regulamento Interno da Direcção do Curso	15
D	Condições de Candidatura	16
E	Programas Resumidos	17
F	Resultados de Aprendizagem por Unidade Curricular	23

1 Enquadramento e Justificação

Originalmente criado pela Portaria nº 420/85, o curso de Licenciatura em Matemática e Ciências de Computação (LMCC) começou a ser ministrado no ano lectivo de 1986/87. Os primeiros licenciados formaram-se no final do ano lectivo de 1990/91. Desde então, licenciados em Matemática e Ciências de Computação têm vindo a integrar quadros de empresas, institutos de investigação e instituições de ensino superior. A continuada procura de licenciados em Matemática e Ciências de Computação pelo mercado das Ciências e Tecnologias da Informação e Comunicação (CTIC), entre outros, tem demonstrado a adequabilidade e o bom nível científico da formação que tem sido ministrada neste curso.

A LMCC foi o segundo curso na área das CTIC a ser criado na Universidade do Minho e, já nessa altura, tinha um perfil bem diferenciado da Licenciatura em Engenharia de Sistemas e Informática. Actualmente, para além dos dois cursos referidos atrás, as licenciaturas da Universidade do Minho nesta área são:

- Licenciatura em Electrónica Industrial e de Computadores;
- Licenciatura em Informática de Gestão;
- Licenciatura em Engenharia de Comunicações.

Neste conjunto de cursos, a LMCC distingue-se pela ênfase dada aos fundamentos teóricos e matemáticos da computação sem descurar o seu cariz científico-tecnológico.

A primeira reestruturação do curso ocorreu no ano lectivo de 1995/96 (Despacho RT/C-185/95), motivada em grande parte pelo avanço científico nas áreas das matemáticas aplicadas e das ciências da computação ao longo de uma década, bem como da experiência pedagógica adquirida ao longo desse período. A segunda reestruturação do curso ocorreu no ano lectivo de 2004/05 (Despacho RT/C-151/2004) e, além das motivações da primeira reestruturação, teve em conta dois factores fundamentais: a Declaração de Bolonha e as conclusões das avaliações feitas pela Sub-Comissão de Avaliação Externa da Fundação das Universidades Portuguesas.

O Decreto Lei 42/2005, de 22 de Fevereiro, abre as portas à criação e reformulação de cursos no espírito da Declaração de Bolonha, estabelecendo a existência de três ciclos de formação no Ensino Superior. Em particular, considera-se que a formação de 1º ciclo é tendencialmente mais generalista e orientada para o acesso a cursos de 2º ciclo, devendo contudo garantir algumas competências para inserção na vida activa.

Recentemente (Dezembro de 2005) o Conselho de Reitores das Universidades Portuguesas (CRUP) aprovou o documento “Ensino Superior: Ordenamento da Oferta Educativa”, que fixa as designações a adoptar em cursos de 1º ciclo.

Neste contexto, é apresenta-se neste documento a proposta de destacar a formação básica ministrada na Licenciatura em Matemática e Ciências de Computação (correspondente aos três primeiros anos do curso) num curso de 1º ciclo, com a designação de Licenciatura em Ciências da Computação (LCC), por esta ser de entre as designações previstas no documento CRUP a que melhor se ajusta à formação ministrada. A formação oferecida actualmente a nível dos quarto e quinto anos de LMCC, manter-se-á no futuro. De momento, os moldes em que esta oferta formativa será disponibilizada encontra-se em equação, dentro de um mapa mais amplo de oferta formativa de segundos ciclos em Ciências e Tecnologias da Informação e Comunicação.

2 Objectivos

As técnicas de computação têm, no mundo moderno, um papel de uma relevância indiscutível. Esse papel tem vindo a ser reforçado não só pelo aumento do espectro de actividades que abrangem (internet, distribuição, segurança, integridade, ...) mas principalmente pelo rigor que nelas é posto. Este rigor só pode ser atingido quando estas técnicas adquirirem uma sólida fundamentação matemática.

Com a formação em Ciências da Computação procura-se dotar os alunos de um conhecimento sólido e extenso dos problemas relacionados com a eficaz e correcta utilização dos métodos e técnicas computacionais. A formação matemática ministrada, por um lado, torna os alunos capazes de desenvolver e verificar rigorosamente soluções correctas e eficientes. Por outro lado, esta formação deverá desenvolver nos alunos uma grande flexibilidade e uma grande adaptabilidade na resolução de novos problemas, levando-os a desenvolver, quando necessário, os seus próprios modelos, métodos, técnicas ou ferramentas computacionais.

O sucesso da formação em CTIC alicerçada nos fundamentos teóricos da computação é atestado pela facilidade de inserção no mercado de trabalho dos licenciados em Matemática e Ciências de Computação e pela opinião dos docentes, ex-alunos de LMCC e empregadores na área das CTIC. Citando o relatório da última avaliação de LMCC

[...]Estes aspectos fazem da LMCC uma licenciatura de sucesso, no panorama nacional.

3 Resultados Esperados da Aprendizagem

Após a conclusão do 1º ciclo de formação universitária em Ciências da Computação, um licenciado deverá ter adquirido as seguintes competências (capacidades cognitivas e práticas, e aptidões):

1. Capacidade de demonstrar conhecimentos e compreensão de conceitos, fundamentos, teorias e factos relacionados com Ciências da Computação e suas aplicações, nomeadamente em algoritmos, estruturas de dados, linguagens e métodos de programação, métodos numéricos, bases de dados, e tecnologias de sistemas de computação e de comunicação.
2. Aptidões no cálculo e raciocínio matemáticos e na construção de argumentos rigorosos, incluindo provas formais.
3. Capacidade de analisar, conceber e implementar algoritmos para a resolução de problemas computacionais, com especial ênfase em critérios de correcção e de eficácia e tomando em consideração os aspectos sociais e éticos relevantes.
4. Capacidade de fundamentadamente seleccionar as ferramentas matemáticas e informáticas mais adequadas à resolução de um dado problema.
5. Capacidade de integração em equipas de desenvolvimento de produtos e serviços de computação.
6. Capacidade efectiva de apresentação e documentação de informação, ideias, problemas e projectos realizados, tanto a públicos constituídos por especialistas como por não especialistas.
7. Capacidade de aprendizagem autónoma de conhecimentos e competências novas ao longo da vida e em complemento às adquiridas.

4 Perfil de Formação

O perfil de formação em Ciências da Computação é marcadamente científico. Esta formação organiza-se em torno do fenómeno da computação e dos formalismos que o fundamentam, lançando assim as bases para uma abordagem rigorosa, produtiva e mesmo original, à resolução de problemas computacionais. Por um lado, são estudadas teorias matemáticas usadas na modelação de diversos aspectos computacionais, incluindo os próprios modelos abstractos de computação. Nesta linha destacam-se estudos em Álgebra, Lógica e Matemática Discreta. Por outro lado, são ensinados os formalismos, em geral de apreciável nível de abstracção, que estão na base de abordagens e tarefas diversas relacionadas com a actividade de programação. Aqui incluem-se paradigmas e métodos de programação, bem como o estudo sob múltiplos pontos de vista do conceito fundamental de algoritmo.

Durante o curso os alunos são ainda munidos de um conjunto base de tecnologias da computação (Arquitetura de Computadores, Comunicação de Computadores, Sistemas Operativos e Bases de Dados) e de técnicas matemáticas de cariz mais aplicacional (nas áreas da Análise Numérica, Probabilidades e Teoria de Números). Estas tecnologias, aliadas a um conjunto base de competências transversais –desenvolvidas essencialmente através da realização de projectos individuais e em grupo, de diversos graus de complexidade–, deverão permitir aos alunos desenvolver competências mínimas para uma razoável inserção imediata na vida activa, em especial na execução de tarefas concretas em projectos de desenvolvimento de produtos e serviços de computação.

5 Estrutura do Curso e Plano de Estudos

A recente reorganização da LMCC assenta na estruturação dos nove semestres da licenciatura em três fases, correspondentes a estádios de formação diferentes:

Básica constituída pelos três primeiros anos;

Intermédia formada pelo quarto ano;

Profissionalizante composta pelo estágio curricular no último semestre.

Um das razões fundamentais para esta reorganização foi o reconhecimento de que as várias licenciaturas em CTIC diferem essencialmente no foco dado na formação básica, partilhando em grande parte a formação intermédia. A proposta que agora se apresenta mantém esta visão, propondo que a formação de 1º ciclo em Ciências da Computação corresponda à formação básica de LMCC. Esta proposta permite aliás concretizar uma das ideias tida como importante para tirar partido das potencialidades desta estruturação de LMCC: a segunda fase de formação não deve ser iniciada sem que antes o aluno complete a formação básica.

Uma das maiores debilidades apontada a LMCC (partilhada pela generalidade das licenciaturas em CTIC) é a decepcionante percentagem de alunos que terminam o curso. Uma das causas deste insucesso deve-se à avidez do mercado de emprego, capaz de absorver eficazmente alunos que não completaram a sua formação universitária. O destacar da formação básica de LMCC, num 1º ciclo conferente de grau, poderá constituir um instrumento importante no combate à tendência de abandono precoce da formação universitária em CTIC.

O curso de LCC centra-se nas teorias e técnicas, já bem estabelecidas, que estão na base das Ciências da Computação. No desenho proposto para o curso, há uma preocupação particular em balancear o peso relativo das unidades curriculares leccionadas pelos dois departamentos específicos do curso. A procura deste equilíbrio reflecte a própria filosofia do curso: uma licenciatura em CTIC com uma sólida fundamentação matemática.

CARACTERIZAÇÃO DO CURSO de LCC	
Estabelecimento de ensino:	Universidade do Minho
Unidade orgânica:	a) Escola de Ciências b) Escola de Engenharia
Curso:	Ciências da Computação
Grau:	Licenciado
Áreas científicas predominantes:	a) Matemática b) Ciências da Computação
Número de créditos necessário à obtenção do grau:	180 ECTS
Duração normal:	6 semestres

Tabela 1: Caracterização do Curso de Licenciatura em Ciências da Computação

A caracterização do curso de acordo com os parâmetros fixados pela Direcção Geral do Ensino Superior é apresentada na Tabela 1. A estrutura curricular do curso, com a identificação do número mínimo de créditos a reunir em cada área científica para a obtenção do grau, apresenta-se na Tabela 2. O plano de estudos da Licenciatura em Ciências da Computação apresenta-se na Tabela 3. Os conteúdos programáticos e o esforço discente por resultado de aprendizagem, relativos às unidades curriculares que compõe o plano de estudos da LCC, apresentam-se, ordenados por semestre, nos Anexos E e F, respectivamente.

O plano de estudos que agora se propõe para LCC corresponde na íntegra aos três primeiros anos do actual plano de estudos de LMCC, tendo sido efectuadas alterações ligeiras a nível da carga lectiva, tipologia de aulas e número de créditos de algumas unidades curriculares. Estas alterações ocorrem nos 1º, 5º e 6º semestres e derivam quer da necessidade de permitir a partilha de unidades curriculares de LCC com a Licenciatura em Matemática quer da experiência já adquirida da leccionação de LMCC no seu formato actual.

A experiência pedagógica anterior demonstrou que à entrada do Ensino Superior os alunos fazem um esforço de adaptação a um novo paradigma de ensino/aprendizagem, o que por si só é um factor de dificuldade que deve de alguma forma ser compensado. Por esta razão, e à semelhança da organização actual de LMCC, o 1º Semestre, embora mantendo o número de créditos ECTS dos outros semestres, é reduzido a quatro unidades curriculares com uma carga

ESTRUTURA CURRICULAR da LCC		
Áreas científicas e créditos necessários à obtenção do grau:		
Cód	Área científica	ECTS
M	Matemática	80
CC	Ciências da Computação	67
TC	Tecnologias da Computação	33
	Total	180

Tabela 2: Estrutura Curricular da Licenciatura em Ciências da Computação

lectiva comparativamente baixa: sendo os créditos ECTS uma medida do esforço discente, estes devem incluir o esforço de adaptação a um novo paradigma de ensino/aprendizagem.

As metodologias de ensino e avaliação a adoptar no curso serão diversas, procurando-se naturalmente que em cada caso sejam encontrados os métodos que mais eficazmente permitam atingir e avaliar os resultados esperados da aprendizagem. A este respeito salientam-se quatro aspectos. Primeiro, é de relevar o forte recurso a aulas práticas laboratoriais como forma de garantir uma experimentação aprofundada de diversas ferramentas e técnicas computacionais. Segundo, tendo em vista criar nos alunos a indispensável autonomia ao futuro exercício de actividade profissional, os alunos deverão desenvolver projectos de programação extra-aulas de vários graus de complexidade. Terceiro, à semelhança do que em pequena escala tem vindo já a ser experimentado na leccionação do 1º Ano do actual plano de estudos de LMCC, deverá ser promovida a integração entre conteúdos programáticos de unidades curriculares distintas, quer através da realização de projectos integrados, quer através da efectiva utilização de saberes ou técnicas aprendidas noutras unidades curriculares. Quarto, é importante não perder a flexibilidade permitida pelo “Modelo de Bolonha” e prever que uma unidade curricular possa funcionar com uma tipologia lectiva semanal não homogénea ao longo do semestre (por exemplo, com maior número de aulas teóricas nas primeiras semanas do semestre e conseqüente redução do número de aulas teóricas nas últimas semanas do semestre). Tais variações ao funcionamento semanal homogéneo, que poderão até envolver várias unidades curriculares, devem ser coordenadas e autorizadas pela Direcção de Curso.

6 Recursos Humanos e Materiais

A presente proposta não pressupõe a aquisição de quaisquer recursos, humanos ou materiais, para ser implementada.

Os quadros docentes dos departamentos envolvidos neste curso têm vindo a estabilizar e as disciplinas novas que foram introduzidas estão bem integradas nas competências deste corpo docente.

Do ponto de vista de laboratórios, tanto o Departamento de Informática como o de Matemática têm feito um esforço substancial de forma a terem boas instalações laboratoriais, bem adaptadas às disciplinas leccionadas.

7 Saídas Profissionais

No mercado de emprego, a área das CTIC é uma das mais atractivas e tem a capacidade de atrair alunos que ainda se encontram a completar a sua formação universitária. Ao completar o 1º ciclo de

PLANO DE ESTUDOS da LCC						
Unidade Curricular	Área	Dep	Horas de Contacto			ECTS
			T	TP	PL	
Álgebra Linear	M	DMat	42	42	0	8
Matemática Computacional	M	DMat	28	0	42	7
Programação Funcional	CC	DI	28	14	28	8
Tópicos de Matemática	M	DMat	28	42	0	7
Total 1º Semestre			294			30
Arquitectura de Computadores	TC	DI	28	0	28	5
Cálculo	M	DMat	42	42	0	7
Matemática Discreta	M	DMat	28	42	0	6
Programação Imperativa	CC	DI	28	14	28	6
Teoria das Linguagens	CC	DMat	28	28	0	6
Total 2º Semestre			336			30
Algoritmos e Complexidade	CC	DI	28	14	28	6
Análise	M	DMat	42	42	0	7
Comunicações por Computador	TC	DI	28	0	28	5
Estruturas Algébricas	M	DMat	42	42	0	7
Lógica	M	DMat	28	28	0	5
Total 3º Semestre			350			30
Cálculo de Programas	CC	DI	28	14	28	6
Lógica Computacional	CC	DI	28	14	28	6
Processamento de Linguagens e Compiladores	TC	DI	28	14	28	6
Programação Orientada aos Objectos	CC	DI	28	14	28	6
Sistemas Operativos	TC	DI	28	0	28	6
Total 4º Semestre			336			30
Análise Numérica	M	DMat	28	14	28	7
Bases de Dados	TC	DI	28	14	14	5
Computabilidade	CC	DMat	28	28	0	6
Geometria	M	DMat	28	28	0	6
Programação Concorrente	CC	DI	28	0	28	6
Total 5º Semestre			294			30
Computação Gráfica	TC	DI	28	14	14	6
Processos e Concorrência	CC	DI	28	28	0	6
Semântica da Programação	CC	DI	28	28	0	5
Teoria das Probabilidades	M	DMat	28	14	28	7
Teoria dos Números Computacional	M	DMat	28	28	0	6
Total 6º Semestre			294			30

Tabela 3: Plano de Estudos da Licenciatura em Ciências da Computação

formação em Ciências da Computação, os alunos deverão ter adquirido um conjunto de competências mínimas que lhes permitam integrar equipas de produção de *software* ou desempenhar tarefas de técnico informático em empresas ou outra organizações. A forte componente fundamental do curso torna os licenciados em Ciências da Computação especialmente aptos para prosseguirem estudos num leque alargado de cursos de 2º ciclo.

8 Encargos

A estrutura curricular, plano de estudos e modelo de funcionamento proposto para o curso de LCC não trará, em princípio, encargos acrescidos quer para a Universidade do Minho, quer para os departamentos envolvidos, quando comparados com os que decorrem do funcionamento actual dos três primeiros da LMCC. O curso proposto corresponde a uma reformulação de um curso já existente, que está já a funcionar em regime estável há vários anos, não havendo portanto necessidade de recursos adicionais e extraordinários. A mudança de paradigma de aprendizagem (essencialmente centrado no aluno) tem associada uma redução dos tempos de contacto actualmente praticados em Portugal. Por outro lado, é indicado que um estudante em regime de tempo integral deverá dedicar 42 horas por semana aos seus cursos. Por outras palavras, será expectável que os estudantes passem mais tempo no campus envolvidos em actividades de estudo. Estas alterações poderão requerer de alguma reconfiguração dos espaços actualmente existentes, designadamente através da criação de espaços específicos para os estudantes poderem trabalhar, quer individualmente, quer em grupo.

9 Calendarização, Processo de Transição e Equivalências

O curso de Licenciatura em Ciências da Computação deve entrar em funcionamento no ano lectivo 2006/2007. Os processos de transição e extinção da Licenciatura em Matemática Ciências da Computação regem-se pelas seguintes regras:

1. Os alunos do Plano Actual de LMCC (PActLMCC) transitam, por omissão, para a Licenciatura em Ciências da Computação. Por esta razão entram simultaneamente em funcionamento, em 2006/2007, os três anos curriculares que compõe o curso.
2. Os alunos inscritos no PActLMCC podem optar por permanecer neste plano, desde que o ano curricular em que pretendem inscrever-se esteja ainda em funcionamento. Os anos curriculares em funcionamento para os diferentes planos de estudo, no presente e nos próximos anos lectivos, indicam-se na Tabela 4 (onde PLCC representa o Plano da Licenciatura em Ciências da Computação e PAntLMCC representa o Plano Antigo de LMCC, que tem vindo a ser progressivamente extinto desde 2004/2005). Por exemplo, o 5º Ano do PActLMCC funcionará pela última vez no ano lectivo de 2009/2010. Este conjunto de regras permite que um qualquer aluno do Plano Actual de LMCC possa concluir este plano, se ainda o desejar, desde que o faça dentro da duração normal do plano.
3. Os alunos do PAntLMCC podem requerer o grau de Licenciatura em Ciências da Computação, desde que tenham completado os três primeiros anos do PAntLMCC. A classificação final é dada pela média das disciplinas que compõem os três primeiros anos do PAntLMCC, ponderada (por unidades de crédito) de acordo com este último plano de estudos.
4. Os alunos do PAntLMCC que não tenham completado os três primeiros anos do curso e que desejem transitar para a Licenciatura em Ciências da Computação devem fazer uma transição prévia para o PActLMCC, aplicando-se depois as regras de transição definidas para os alunos do PActLMCC.

Ano lectivo	Ano curricular				
	1º	2º	3º	4º	5º
2005/2006	PActLMCC	PActLMCC	PAntLMCC	PAntLMCC	PAntLMCC
2006/2007	PLCC	PLCC + PActLMCC	PLCC + PActLMCC	PAntLMCC	PAntLMCC
2007/2008	PLCC	PLCC	PLCC + PActLMCC	PActLMCC	PAntLMCC
2008/2009	PLCC	PLCC	PLCC	PActLMCC	PActLMCC
2009/2010	PLCC	PLCC	PLCC	–	PActLMCC
≥ 2010/2011	PLCC	PLCC	PLCC	–	–

Tabela 4: Calendarização

Os seis primeiros semestres do plano actual de LMCC coincidem com o plano de estudos de Ciências da Computação, ao nível das unidades curriculares, incluindo designações e semestres de leccionação. Deste modo, estabelece-se que cada unidade curricular do plano de Ciências da Computação é equivalente à unidade curricular do plano actual de LMCC com idêntica designação. Assim, ao transitar para a Licenciatura em Ciências da Computação, um aluno inscrito num dos três primeiros anos curriculares do plano actual de LMCC mantém a sua situação escolar.

ANEXOS

A Minuta de Resolução do Senado Universitário

Resolução SU – ____/2____

Sob proposta das Escolas de Ciências e de Engenharia:

Ouvido o Conselho Académico, nos termos da alínea g) do n.º 2 do artigo 24.º dos Estatutos da Universidade do Minho;

Ao abrigo do disposto no n.º 1 do artigo 7.º da Lei n.º 108/88, de 24 de Setembro; no n.º 1 do artigo 1.º do Decreto-Lei n.º 155/89, de 11 de Maio; no Decreto-Lei n.º 42/2005, de 22 de Fevereiro; e no n.º 2 do artigo 20.º dos Estatutos da Universidade do Minho:

O Senado Universitário da Universidade do Minho, em sessão plenária de ____ de ____ de 2____, determina:

1.º

(Criação de curso)

A Universidade do Minho passa a conferir o grau de licenciado em Ciências da Computação, ministrando, em consequência, o respectivo curso.

2.º

(Organização do curso)

O curso de Licenciatura em Ciências da Computação, da Universidade do Minho, adiante simplesmente designado por curso, organiza-se pelo sistema de unidades de crédito europeus (ECTS).

3.º

(Estrutura curricular)

A estrutura curricular do curso consta em anexo à presente Resolução.

4.º

(Plano de estudos)

O plano de estudos do curso será fixado por despacho do Reitor, sob proposta do Conselho Académico, a publicar na II série do *Diário da República*.

5.º

(Precedências)

As tabelas e o regime de precedências serão fixados por despacho do Reitor, sob proposta do Conselho Académico.

6º

(Classificação final)

1. A classificação final do curso é a média aritmética ponderada, arredondada às unidades (considerando como unidade a fracção não inferior a cinco décimas), das classificações das unidades curriculares em que o aluno realizou os créditos necessários à satisfação do disposto no anexo a esta Resolução.
2. Os coeficientes de ponderação serão fixados por despacho do Reitor, sob proposta do Conselho Académico.
3. Paralelamente, será atribuída a cada aluno uma classificação de acordo com a escala europeia de comparabilidade (de A a E), a qual é estabelecida pelo Conselho Académico.

7º

(Condições de acesso)

As condições de acesso, matrícula, inscrição, reingresso, transferência e mudança de curso são as fixadas anualmente para os cursos do 1º ciclo da Universidade do Minho, observando o disposto sobre a matéria no Decreto-Lei nº 296-A/98, de 25 de Setembro, com as alterações introduzidas pelo Decretos-Lei nº 99/99, de 30 de Março, 26/2003, de 7 de Fevereiro, 76/2004, de 27 de Março, e 158/2004, de 30 de Junho.

8º

(Calendário escolar)

A duração dos períodos lectivos será a que, nos termos da alínea b), nº 2, artigo 24º dos Estatutos, for fixada no calendário escolar da Universidade do Minho.

9º

(Início do funcionamento)

O curso terá início a partir de 2____/2____ em diante.

Universidade do Minho, ____ de _____ de 2____

O Presidente do Senado Universitário

Anexo

Resolução SU – _____/2_____

1. Áreas científicas do curso

- Matemática
- Ciências da Computação
- Tecnologias da Computação

2. Duração normal do curso

6 semestres

3. Número de unidades de crédito necessários para a obtenção do grau

180 créditos (ECTS)

4. Áreas científicas e distribuição das unidades de crédito

Áreas científicas obrigatórias

Matemática	80	créditos (ECTS)
Ciências da Computação	67	créditos (ECTS)
Tecnologias da Computação	33	créditos (ECTS)

Áreas científicas opcionais

Não se aplica.

5. Taxa de matrícula e propinas

Estes montantes serão fixados pelo Conselho Académico nos termos dos Estatutos da Universidade do Minho.

B Ficha relativa ao enquadramento no Conselho de Cursos

Licenciatura em Ciências da Computação

1. Conselho de Cursos em que se integra

- Conselho de Cursos de Ciências

2. Departamentos específicos (nº 2 do art. 44º dos Estatutos da Universidade do Minho)

- Departamento de Informática
- Departamento de Matemática

3. Departamentos não específicos

Não se aplica.

C Proposta de Regulamento Interno da Direcção do Curso

Propõe-se que a Comissão de Curso tenha a seguinte constituição:

- Director do Departamento de Informática ou seu representante;
- Director do Departamento de Matemática ou seu representante;
- Um representante do Departamento de Informática e um do Departamento de Matemática;
- Três representantes dos alunos, sendo um por cada ano do curso.

O cargo de Director de Curso deverá ser rotativamente exercido pelos directores dos departamentos específicos ou seu representantes, por períodos de 2 anos.

Durante o período de co-existência de LMCC e de LCC, o Director de Curso de LMCC deve acumular as funções de director de curso de LCC.

D Condições de Candidatura

Os alunos candidatos à Licenciatura em Ciências da Computação da Universidade do Minho deverão ter efectuado prova de ingresso (16-Matemática) com classificação mínima de 95 (num total de 200 pontos).

E Programas Resumidos

1º Semestre

Programação Funcional

Programa: Introdução ao paradigma funcional; a linguagem de programação Haskell. Expressões, valores e redução. Tipos básicos e tipos algébricos. Funções recursivas e tipos de dados recursivos. Programação com listas, árvores binárias e árvores binárias de pesquisa. Programação funcional de ordem superior. Polimorfismo; classes; tipos principais. Modularidade.

Álgebra Linear

Programa: Matrizes: operações com matrizes, inversa de uma matriz; transposição. Sistemas de equações lineares: o algoritmo de eliminação de Gauss, o algoritmo de Gauss-Jordan para inversão de matrizes; regra de Cramer. Espaços vectoriais reais e complexos: subespaços, dependência e independência lineares; base e dimensão. Transformações lineares: transformações lineares entre espaços vectoriais de dimensão finita, matriz de uma transformação linear, núcleo e imagem de uma transformação linear, nulidade e característica. Determinantes: propriedades; teorema de Laplace. Valores e vectores próprios: valores e vectores próprios em espaços vectoriais de dimensão finita, subespaços próprios, diagonalização.

Matemática Computacional

Programa: Introdução à Matemática Computacional. Apresentação do Mathematica®: capacidades numéricas, capacidades simbólicas, funções, listas e capacidades gráficas. Elementos de programação. Breve exploração de um sistema algébrico computacional “freeware”.

Tópicos de Matemática

Programa: Introdução ao cálculo proposicional: conectivos, fórmulas, valores de verdade, valorações, tautologias, equivalências lógicas. Representação de conjuntos, relações binárias, aplicações, famílias de conjuntos, relações de equivalência, conjuntos quociente. Conjuntos finitos e conjuntos infinitos, conjuntos numeráveis e conjuntos não numeráveis, teorema de Cantor. Princípio de indução e princípio de indução completa. Conjuntos parcialmente ordenados: elementos especiais, conjuntos bem ordenados.

2º Semestre

Programação Imperativa

Programa: Introdução ao paradigma imperativo; representação de dados; estruturas de controlo; manuseamento de ficheiros; gestão dinâmica de memória; estruturas de dados lineares; pesquisa e ordenação: conceitos e algoritmos; tipos de dados abstractos: pilhas e filas de espera; estruturas de dados não lineares: árvores; árvores binárias de pesquisa. (suporte: linguagem C).

Teoria das Linguagens

Programa: Linguagens; expressões regulares e linguagens regulares; autómatos finitos deterministas e autómatos finitos não deterministas; equivalência entre os dois tipos de autómatos; teorema de Kleene; minimização de autómatos; lema da bombagem; gramáticas independentes do contexto; autómatos de pilha; correspondência entre gramáticas independentes de contexto e autómatos de pilha.

Cálculo

Programa: Axiomática do corpo dos números reais. Sucessões e séries numéricas, critérios de convergência. Continuidade de funções reais de variável real. Teoremas de Weierstrass e do Valor Intermédio. Função composta e função inversa. Derivabilidade, teoremas de Rolle, Lagrange e Cauchy, Regra de l'Hôpital. Funções hiperbólicas, hiperbólicas inversas e trigonométricas inversas. Primitivas imediatas, por partes, por substituição e primitivação de fracções racionais. Integral de uma função limitada num intervalo. Propriedades do integral. Teorema da mudança de variável. O Teorema Fundamental do Cálculo e algumas consequências.

Matemática Discreta

Programa: Noções básicas sobre grafos: caminhos, atalhos, componentes conexas. Grafos eulerianos e grafos hamiltonianos. Grafos planares, sua caracterização. Fórmula de Euler. Introdução à teoria de números: divisibilidade, números primos, máximo divisor comum e mínimo múltiplo comum, algoritmo de Euclides, teorema fundamental da aritmética. Equações diofantinas, congruências, o anel Z_n , teorema chinês dos restos. Função e fórmula de Euler, teorema de Wilson.

Arquitectura de Computadores

Programa: Organização e estrutura de um computador; representação de informação; arquitectura ao nível do conjunto de instruções; avaliação de desempenho de computadores.

3º Semestre

Comunicações por Computador

Programa: Transmissão e comunicação de dados; protocolos de comunicações; famílias de protocolos e suas arquiteturas; redes locais de computadores; protocolos TCP/IP.

Estruturas Algébricas

Programa: Grupos: subgrupos e subgrupos invariantes, Teorema de Lagrange, grupos quociente, homomorfismos. Anéis: subanéis, ideais, anel quociente, homomorfismos; domínios de integridade; corpos. Reticulados: subreticulados, homomorfismos e produto directo; relações de congruência e reticulados quociente. Reticulados distributivos e reticulados modulares. álgebras de Boole. Conceitos básicos em algebra universal: operação de aridade n , álgebras; subálgebras, homomorfismos e produto directo; subálgebra gerada; relações de congruência e álgebra quociente.

Algoritmos e Complexidade

Programa: Introdução à análise de algoritmos: análise de correcção; análise assintótica de tempo de execução; Notações \mathcal{O} , Ω , Θ ; relações de recorrência; estratégias algorítmicas fundamentais. Algoritmos de ordenação. Algoritmos clássicos sobre grafos. Definição informal das classes de problemas P e NP; exemplos de problemas NP-completos. Estruturas de dados: árvores AVL e tabelas de “hash”. (suporte: linguagem C).

Análise

Programa: Produto interno, norma e distância. Conjuntos abertos, fechados, limitados, compactos e conexos por arcos em \mathbb{R}^n . Sucessões em \mathbb{R}^n . Funções vectoriais, limites e continuidade, continuidade de funções compostas. Noção de derivada direccionada, derivada parcial. Generalização do Teorema de Lagrange para funções reais de várias variáveis. Derivada (global). Derivadas de ordem superior. Derivada da função composta. Hiperplano tangente ao gráfico de uma função num ponto onde a função é derivável. Recta normal e hiperplano tangente a uma curva de nível. Máximos e mínimos locais e condicionados. Teorema da Função Inversa e Teorema da Função Implícita. Breve introdução à definição de integral de uma função definida num rectângulo. Extensão do conceito de integral a outros conjuntos. Teorema de Fubini. Teorema da mudança de variável. Coordenadas polares, cilíndricas e esféricas.

Lógica

Programa: Sintaxe e semântica da Lógica Clássica Proposicional; formas normais conjuntivas e disjuntivas; sistema formal de dedução natural; teoremas da correcção e completude. Sintaxe, semântica de Tarski e dedução natural para a Lógica Clássica de Primeira Ordem. Introdução à Lógica Intuicionista Proposicional: semântica de Kripke; dedução natural; interpretação de Gödel em Lógica Clássica.

4º Semestre

Programação Orientada aos Objectos

Programa: Introdução à programação por objectos: origem do paradigma e conceitos básicos fundamentais. Noção de objecto em PPO: estrutura e comportamento, encapsulamento, mensagens e métodos, instâncias e classes. Classes, hierarquias de classes e herança: herança lógica vs herança de implementação, herança vs agregação. Classes abstractas: reutilização e extensibilidade, tipos abstractos de dados, polimorfismo. Concepções de sistemas em PPO: subclasses como especialização, tipos de dados estáticos e dinâmicos. Programação por objectos utilizando Java: modelo de programação em Java, estudo das principais bibliotecas, “static checking” vs “runtime checking”.

Cálculo de Programas

Programa: Arquitectura da programação: teoria e método. Composicionalidade. Programação funcional composicional, seus combinadores e suas leis de cálculo. Programação funcional com mónadas e o seu cálculo. Estruturas de dados indutivas regulares e sua álgebra de programação. Programação genérica: parametrização e polimorfismo. Customização e reutilização.

Processamento de Linguagens e Compiladores

Programa: Análise e geração automática de programas; gramáticas independentes do contexto; “parsing top-down” e “parsing bottom-up”; programação baseada em gramáticas de atributos.

Lógica Computacional

Programa: Lógicas modais; Lógica Linear; sistemas dedutivos; “tableaux”; sistemas dedutivos otimizados para o cálculo proposicional. Programação lógica.

Sistemas Operativos

Programa: Objectivos, evolução e estrutura dos sistemas operativos modernos; gestão de processos (objectivos, critérios e estratégias de escalonamento); gestão de memória (memória real e virtual, paginação e segmentação); gestão de periféricos (“software” de I/O, gestão de disco, “device drivers”); sistemas de ficheiros.

5º Semestre

Computabilidade

Programa: Máquinas de Turing; linguagens recursivas e recursivamente enumeráveis; máquinas de Turing universais; problemas de decisão; problemas (in)decidíveis e problemas semi-decidíveis; problemas de decisão em teoria da computabilidade: problema da auto-rejeição, problema da aceitação e problema da paragem, teorema de Rice; reduções entre problemas; funções computáveis por máquinas de Turing, funções parciais recursivas e equivalência entre as duas classes de funções; Tese de Church. Introdução à Teoria da Complexidade; classes P e NP; completude NP; teorema de Cook e outros exemplos de problemas completos para NP.

Análise Numérica

Programa: Erros e estabilidade. Interpolação polinomial. Quadratura: fórmulas de Newton-Cotes. Equações não lineares. Sistemas de equações lineares: métodos directos. Utilização de um sistema computacional na programação de métodos estudados.

Geometria

Programa: Estrutura euclidiana de \mathbb{R}^2 e \mathbb{R}^3 : rectas, planos, sistemas de referência, produto interno e distância, paralelismo e perpendicularidade. Transformações geométricas de \mathbb{R}^2 e \mathbb{R}^3 : isometrias no plano e no espaço; semelhanças e inversões no plano. Geometria projectiva: coordenadas homogéneas, perspectivas, projecções paralelas. Curvas e superfícies: parametrizações de curvas e superfícies.

Bases de Dados

Programa: Terminologia de sistemas de bases de dados; análise, planeamento e concepção de sistemas de bases de dados; manipulação de dados; sistemas de bases de dados distribuídas e centralizadas.

Programação Concorrente

Programa: Modelação de sistemas concorrentes: processos, estados, acções, escolha, não determinismo, guardas, propriedades de segurança e progresso; Concorrência em memória partilhada: exclusão mútua, “deadlocks”, semáforos, monitores (tipos abstractos de dados concorrentes, variáveis de condição, espera e semânticas de sinalização, invariantes e predicados, concorrência em linguagens orientadas a objectos); Concorrência em sistemas distribuídos: canais e portos, “send” e “receive”, “unicast” e “broadcast”, modelos síncronos e assíncronos, modelo cliente-servidor, faltas em processos e canais, semântica perante faltas, tempo, ordem e causalidade.

6º Semestre

Processos e Concorrência

Programa: Introdução à modelação formal e verificação de sistemas concorrentes: sistemas de transição de estados; propriedades de segurança, animação e justiça; introdução ao cálculo de processos. Redes de Petri: modelação de sistemas concorrentes; semântica operacional; cálculo de invariantes; introdução às redes coloridas; ferramentas para especificação e animação de redes de Petri. Lógica Temporal: especificação de propriedades de sistemas concorrentes usando lógica CTL; verificação directa de modelos; verificação simbólica de modelos baseada em OBDDs; ferramentas para verificação simbólica de modelos.

Semântica da Programação

Programa: O λ -calculus: sintaxe, redução, avaliação (ordens normal e applicativa). Estudo de uma linguagem de programação funcional elementar: semântica operacional e semântica denotacional. Introdução à teoria de domínios: funções contínuas e menores pontos fixos. Semântica de funções recursivas. O sistema de tipos simples: tipos implícitos e explícitos; inferência de tipos.

Teoria dos Números Computacional

Programa: Raízes primitivas e índices, aplicação ao estudo de alguns tipos de congruências. Distribuição dos números primos, teorema dos números primos. Testes de primalidade e de pseudo-primalidade. Algoritmos para gerar números primos. Código RSA. Resíduos quadráticos, lei da reciprocidade quadrática. Frações racionais, equações de Pell. Algoritmos de factorização (Fermat, Pollard, via frações contínuas, etc.). Representação de inteiros como soma de quadrados. Inteiros e primos gaussianos.

Teoria das Probabilidades

Programa: Introdução e conceitos básicos. Teoria axiomática. Variáveis aleatórias, vectores aleatórios e distribuições de probabilidade (discretas, absolutamente contínuas e outras). Medidas de localização, dispersão e forma. Momentos, desigualdades e transformadas. Famílias de distribuições univariadas e multivariadas mais comuns. Funções de variáveis aleatórias. O modelo normal e suas propriedades. Distribuições por amostragem em modelos normais. Convergências estocásticas. Teorema Limite Central e leis dos grandes números. Simulação computacional (ao longo do programa).

Computação Gráfica

Programa: Estudo das primitivas gráficas; transformações geométricas (escala, rotação e translação); cores e texturas; iluminação; som; análise do modelo de eventos; interpoladores: animação básica; encaminhamento de eventos; técnicas avançadas para melhorar o desempenho; protótipos.

F Resultados de Aprendizagem por Unidade Curricular

Nas folhas que se seguem, apresentam-se os resultados esperados de aprendizagem, créditos ECTS e horas de trabalho por tarefa relativos às unidades curriculares que compõe o plano de estudos da Licenciatura em Ciências da Computação.

CURSO: Ciências da Computação
 UNIDADE CURRICULAR: Álgebra Linear

Anexo 2

ÁREA CIENTÍFICA: Matemática (M)

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Distribuição das horas creditadas ao aluno para obtenção de 8 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo	Trab° projecto		
	T	TP	PL	TC	S	OT	E					
Operar com matrizes.	6	6						15				
Compreender demonstrações e argumentações simples de resultados básicos de Álgebra Linear.	10	7						23				
Demonstrar e aplicar propriedades de espaços vectoriais.	8	9						28				
Resolver problemas que envolvam aplicações lineares.	6	7						19				
Determinar a solução geral de um sistema de n equações lineares em m incógnitas.	6	6						15				
Caracterizar aplicações lineares diagonalizáveis.	6	7						24				
TOTAL	42	42						124	12		4	224

CURSO: Ciências da Computação

UNIDADE CURRICULAR: **Matemática Computacional**

ÁREA CIENTÍFICA: **Matemática (M)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 7 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trabº grupo			Trabº projecto
	T	TP	PL	TC	S	OT	E					
Conhecer as regras básicas do Mathematica.	1		2					4				
Pesquisar e utilizar funcionalidades recorrendo aos sistemas de ajuda do Mathematica.	1		1					4				
Resolver de problemas de Análise Real, Álgebra Linear e Teoria de Números com recurso a funcionalidades do Mathematica.	18		27					84				
Construir funções/programas, usando programação funcional e recursiva, que desempenhe tarefas especificadas.	4		6					14				
Utilizar regras de reescrita como elemento de atribuição e de programação.	2		3					8				
Importar e exportar dados de forma estruturada.	2		3					8				
TOTAL	28		42					122			4	196

CURSO: **Ciências da Computação**

UNIDADE CURRICULAR: **Programação Funcional**

ÁREA CIENTÍFICA: **Ciências da Computação (CC)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de **8** créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo			Trab° projecto
	T	TP	PL	TC	S	OT	E					
Definir funções não recursivas e recursivas simples.	6	3	8					17	17			
Compreender o conceito de tipo indutivo.	4	2	4					16	10			
Definir e usar tipos algébricos.	5	3	6					13	14			
Compreender a noção de tipo principal e polimorfismo.	5	2	4					13	10			
Definir e usar funções de ordem superior.	4	2	4					10	10			
Definir e usar classes.	4	2	2					10	10			
TOTAL	28	14	28					80	70		4	224

CURSO: Ciências da Computação

UNIDADE CURRICULAR: Tópicos de Matemática

ÁREA CIENTÍFICA: Matemática (M)

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 7 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Labora- toriais	T. de campo	Seminá rio	Tutórias	Estágios	Estudo	Trabº grupo			Trabº projecto
Listagem de RA (entre 4 e 6)	T	TP	PL	TC	S	OT	E					
Aplicar propriedades elementares das operações lógicas proposicionais e de quantificação.	5	8						22				
Construir argumentos matemáticos usando métodos de prova comum, incluindo indução sobre os naturais.	4	6						18				
Explicar os conceitos básicos sobre conjuntos, relações e funções e realizar operações elementares sobre estes objectos matemáticos.	9	14						40				
Exemplificar os conceitos básicos sobre relações de equivalência e relações de ordem.	7	10						30				
Distinguir entre conjuntos numeráveis e conjuntos não numeráveis.	3	4						12				
TOTAL	28	42						122			4	196

CURSO: Ciências da Computação

UNIDADE CURRICULAR: **Arquitectura de Computadores**

ÁREA CIENTÍFICA: **Tecnologias da Computação (TC)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 5 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo	Trab° projecto		
	T	TP	PL	TC	S	OT	E					
Descrever e manipular informação (textos, imagens, números inteiros/ reais e instruções de CPU), em binário ou em hexadecimal.	4		7					5	4			
Descrever a estrutura e organização de um computador típico e identificar técnicas para melhorar o desempenho dos processadores.	9		7					15	12			
Identificar as características mais relevantes do conjunto de instruções de um CPU e discutir um ISA tipo.	7		7					11	8			
Analisar, interpretar, testar e modificar código <i>assembly</i> de um CPU típico (e.g. IA32), gerado por um compilador (e.g. de C), incluindo estruturas de controlo/dados e invocação de procedimentos/funções.	7		5					13	10			
Assumir atitudes de aprendizagem activa e responsável, de trabalho persistente individual e de grupo, e de aplicação de espírito crítico na análise e resolução de problemas.	1		2									
TOTAL	28		28					44	34		6	140

CURSO: Ciências da Computação

UNIDADE CURRICULAR: Cálculo

ÁREA CIENTÍFICA: Matemática (M)

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 7 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo			Trab° projecto
	T	TP	PL	TC	S	OT	E					
Listagem de RA (entre 4 e 6)												
Compreender as principais noções topológicas e de ordem do corpo dos números reais.	4	4										
Conhecer as principais noções sobre sucessões e séries numéricas.	8	8										
Saber aplicar ao estudo de funções resultados sobre continuidade e derivabilidade.	15	15										
Conhecer os teoremas fundamentais do cálculo e aplicar as técnicas de integração.	15	15										
TOTAL	42	42						108			4	196

CURSO: Ciências da Computação

UNIDADE CURRICULAR: **Matemática Discreta**

ÁREA CIENTÍFICA: **Matemática (M)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATORIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 6 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo	Trab° projecto		
	T	TP	PL	TC	S	OT	E					
Aplicar o Algoritmo de Euclides para o cálculo do máximo divisor comum entre dois inteiros.	5	7						16				
Resolver equações diofantinas.	3	5						12				
Conhecer critérios de primalidade.	3	5						12				
Conhecer a noção de congruência módulo um inteiro e suas propriedades.	4	6						12				
Resolver congruências lineares e sistemas de congruências lineares.	5	7						16				
Reconhecer e caracterizar classes de grafos.	8	12						26				
TOTAL	28	42						94			4	168

CURSO: **Ciências da Computação**

UNIDADE CURRICULAR: **Programação Imperativa**

ÁREA CIENTÍFICA: **Ciências da Computação (CC)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de **6** créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo			Trab° projecto
	T	TP	PL	TC	S	OT	E					
Ser capaz de desenvolver algoritmos para problemas de cálculo numérico.	6	3	6					6	8		1	
Ser capaz de desenvolver algoritmos para problemas envolvendo manipulação de strings.	4	2	4					4	10		1	
Ser capaz de desenvolver algoritmos para problemas envolvendo armazenamento e persistência da informação.	6	3	6					6	12		3	
Ser capaz de desenvolver algoritmos para problemas envolvendo processamento de estruturas de dados (baseadas em co-produtos, listas e árvores).	8	4	8					8	20		1	
Resolver problemas de média dimensão por decomposição.	2	1	2					2	8		1	
Ser capaz de codificar em C qualquer algoritmo que tenha especificado.	2	1	2					2	2		3	
TOTAL	28	14	28					28	60		10	168

CURSO: **Ciências da Computação**

UNIDADE CURRICULAR: **Teoria das Linguagens**

ÁREA CIENTÍFICA: **Ciências da Computação (CC)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de **6** créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo			Trab° projecto
	T	TP	PL	TC	S	OT	E					
Conhecer os conceitos básicos relacionados com linguagens formais, autómatos e gramáticas.	16	10						41				
Identificar linguagens regulares.	1	4						9				
Relacionar as noções de autómatos finitos, linguagens reconhecíveis e linguagens regulares.	4	4						14				
Verificar se uma palavra é aceite por um autómato de pilha.	3	4						14				
Determinar a linguagem gerada por uma gramática livre de contexto e o autómato de pilha que a reconhece.	2	4						14				
Reconhecer os autómatos como modelos abstractos de computação.	2	2						4				
TOTAL	28	28						98	10		4	168

CURSO: Ciências da Computação

UNIDADE CURRICULAR: Algoritmos e Complexidade

ÁREA CIENTÍFICA: Ciências da Computação (CC)

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATORIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 6 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo			Trab° projecto
	T	TP	PL	TC	S	OT	E					
Utilizar estruturas de dados não-lineares eficientes para pesquisa: árvores AVL e tabelas de <i>hash</i> .	2	4	8					10	10			
Utilizar estruturas de dados baseadas em grafos; representar grafos em computador; conhecer e utilizar algoritmos baseados em grafos..	9	3	6					12	20			
Analisar a correcção de algoritmos recorrendo a invariantes de ciclo.	4	2	4					8				
Conhecer e comparar algoritmos clássicos de ordenação.	3	1	2					6				
Analisar assintoticamente o tempo de execução de algoritmos; utilizar técnicas como a análise agregada e as equações de recorrência.	6	3	8					12	10			
Conhecer exemplos de problemas da classe NP; produzir reduções polinomiais de problemas; provar problemas NP-completos.	4	1	0					6				
TOTAL	28	14	28					54	40		4	168

CURSO: **Ciências da Computação**

UNIDADE CURRICULAR: **Análise**

ÁREA CIENTÍFICA: **Matemática (M)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de **7** créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trabº grupo	Trabº projecto		
	T	TP	PL	TC	S	OT	E					
Conhecer as noções básicas da topologia de \mathbb{R}^n .	6	6										
Compreender as noções de continuidade e derivabilidade de funções de várias variáveis.	12	12										
Conhecer e aplicar os teoremas da função inversa e da função implícita.	6	6										
Determinar e classificar os extremos livres e condicionados de funções de várias variáveis.	7	7										
Saber calcular e interpretar integrais múltiplos.	11	11										
TOTAL	42	42						108			4	196

CURSO: **Ciências de Computação**

UNIDADE CURRICULAR: **Comunicações por Computador**

ÁREA CIENTÍFICA: **Tecnologias da Computação (TC)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de **5** créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Laborato-riais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trabº grupo	Trabº projecto		
	T	TP	PL	TC	S	OT	E					
Discutir os conceitos fundamentais sobre comunicação de dados, protocolos de comunicação, famílias de protocolos e sua arquitectura, incluindo os elementos protocolares mais comuns.	6		4					10	8			
Analisar e instanciar esses conceitos, em especial, no nível de ligação de dados.	6		6					8	8			
Ter uma visão global e crítica sobre as redes locais de computadores e do seu funcionamento.	6		8					8	8			
Explicar em detalhe os principais protocolos TCP/IP.	10		10					15	15			
TOTAL	28		28					41	39		4	140

CURSO: **Ciências da Computação**

UNIDADE CURRICULAR: **Estruturas Algébricas**

ÁREA CIENTÍFICA: **Matemática (M)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 7 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trabº grupo			Trabº projecto
	T	TP	PL	TC	S	OT	E					
Compreender a noção de álgebra e, em particular, a noção de operação de aridade n .	4	4						10				
Distinguir algumas classes importantes de álgebras tais como grupos, anéis, reticulados e álgebras de Boole, reconhecendo as suas propriedades fundamentais.	10	11						30				
Reconhecer homomorfismos e interpretar propriedades com eles relacionadas.	8	8						20				
Aplicar operadores sobre álgebras, identificando as álgebras obtidas.	7	7						17				
Identificar álgebras quociente	6	6						14				
Resolver problemas e demonstrar resultados básicos da disciplina.	7	6						17				
TOTAL	42	42						108			4	196

CURSO: **Ciências da Computação**

UNIDADE CURRICULAR: **Lógica**

ÁREA CIENTÍFICA: **Matemática (M)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de **5** créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trabº grupo			Trabº projecto
	T	TP	PL	TC	S	OT	E					
Manipular a sintaxe formal da lógica proposicional e da lógica de 1ª ordem.	7	7						16				
Usar fórmulas da lógica de 1ª ordem para representar frases em linguagem natural.	2	1						4				
Dar significado a fórmulas, e decidir o seu valor lógico, no contexto de uma interpretação.	8	9						22				
Compreender e construir demonstrações formais em dedução natural proposicional e de 1ª ordem.	5	6						18				
Conhecer algumas das consequências dos teoremas da correcção e completude.	3	2						8				
Explicar diferenças entre validade intuicionista e validade clássica.	3	3						10				
TOTAL	28	28						78			6	140

CURSO: Ciências da Computação

UNIDADE CURRICULAR: Cálculo de Programas

ÁREA CIENTÍFICA: Ciências da Computação (CC)

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 6 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trabº grupo	Trabº projecto		
	T	TP	PL	TC	S	OT	E					
Perceber as regras da transformação <i>pointfree</i> .	2	1	2					6	5			
Ser capaz de estabelecer raciocínios equacionais simples sobre programas funcionais.	4	2	4					9	6			
Escrever programas funcionais com recurso a padrões de recursividade do tipo "fold" e "unfold" sobre tipos de dados regulares arbitrários.	6	3	6					11	7			
Ser capaz de estabelecer raciocínios equacionais sobre programas recursivos genéricos nos tipos de dados.	6	3	6					11	7			
Interpretar programas funcionais arbitrários como hilomorfismos.	6	3	6					11	7			
Perceber a programação funcional monádica e seu cálculo.	4	2	4					8	6			
TOTAL	28	14	28					56	38		4	168

CURSO: **Ciências da Computação**

UNIDADE CURRICULAR: **Lógica Computacional**

ÁREA CIENTÍFICA: **Ciências da Computação (CC)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de **6** créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trabº grupo	Trabº projecto		
	T	TP	PL	TC	S	OT	E					
Rever as noções sintácticas e semânticas essenciais da Lógica de Primeira Ordem.	6	3						12				
Compreender o conceito de resolução e de Programação Lógica.	4	2	10					8		12		
Compreender os sistemas dedutivos generalistas da Lógica de 1ª Ordem e otimizados para a Lógica Proposicional.	10	2						20				
Dominar o uso de demonstradores de teoremas generalistas e otimizados.		3	10							12		
Compreender as noções essenciais da Lógica Modal e Linear e dos seus sistemas dedutivos.	6	4						16				
Relacionar Programação Lógica e demonstradores de teoremas.	2		8							12		
TOTAL	28	14	28					56		36	6	168

CURSO: Ciências da Computação

Anexo 2

UNIDADE CURRICULAR: **Processamento de Linguagens e Compiladores**ÁREA CIENTÍFICA: **Tecnologias da Computação (TC)**UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA OBRIGATÓRIA OPCIONAL

Distribuição das horas creditadas ao aluno para obtenção de 6 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Laboratoriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trabº grupo	Trabº projecto		
	T	TP	PL	TC	S	OT	E					
Especificar formalmente uma linguagem através de uma gramática; estruturar um Processador de Linguagens.	3	1	2					6				
Implementar o módulo de Análise Léxica com base em autómatos deterministas.	1	2	4					2				
Implementar o Módulo de Análise Sintáctica, segundo as estratégias Top-Down, ou Bottom-Up.	12	5	10					24				
Desenvolver um Processador de Linguagens segundo o método da Tradução Dirigida pela Sintaxe, usando uma gramática tradutora.	2	2	4					4	15			
Desenvolver um Processador de Linguagens segundo o método da Tradução Dirigida pela Semântica, usando uma gramática de atributos.	6	2	4					12	15			
Desenvolver um Compilador, gerando código assembly de uma máquina de stack virtual.	4	2	4					8	6			
TOTAL	28	14	28					56	36		6	168

CURSO: **Ciências da Computação**

UNIDADE CURRICULAR: **Programação Orientada aos Objectos**

ÁREA CIENTÍFICA: **Ciências da Computação (CC)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de **6** créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Laboratoriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trabº grupo			Trabº projecto
	T	TP	PL	TC	S	OT	E					
Compreender e identificar os conceitos básicos principais da POO.	6	2	4					8	4			
Analisar e compreender a importância da estruturação de um domínio em classes.	4	2	4					8	8			
Compreender a importância das classes abstractas e herança na concepção de um sistema em POO.	6	3	6					10	8			
Construir hierarquias de classes utilizando as noções de classe abstracta, tipo abstracto de dados e polimorfismo.	6	4	6					12	10			
Saber utilizar linguagens de programação orientadas aos objectos para construção de programas.	6	3	8					12	14			
TOTAL	28	14	28					50	44		4	168

CURSO: **Ciências da Computação**

UNIDADE CURRICULAR: **Sistemas Operativos**

ÁREA CIENTÍFICA: **Tecnologias da Computação (TC)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de **6** créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Laborato- riais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo	Trab° projecto		
	T	TP	PL	TC	S	OT	E					
Compreender o papel do sistema operativo num sistema informático.	4		2					6				
Demonstrar conhecimento da arquitectura e operação de sistemas operativos actuais.	4		4					6				
Perceber o conceito de processo/tarefa e a forma como um sistema operativo gere múltiplas actividades.	4		6					9	7	10		
Discutir compromissos, mecanismos e estratégias de gestão de recursos.	8		6					13	7	9		
Escrever programas multi-tarefa simples utilizando correctamente primitivas de sincronização e comunicação.	8		10					15		24		
TOTAL	28		28					49	14	43	6	168

CURSO: **Ciências da Computação**

UNIDADE CURRICULAR: **Análise Numérica**

ÁREA CIENTÍFICA: **Matemática (M)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATORIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 7 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutoriais	Estágios	Estudo	Trab° grupo			Trab° projecto
	T	TP	PL	TC	S	OT	E					
Conhecer a teoria básica dos erros numéricos e a aritmética IEEE.	3	1	2					10				
Conhecer e aplicar a teoria básica de alguns métodos numéricos de interpolação e aproximação polinomial, de quadratura numérica, de solução equações não-lineares e de sistemas de equações .	18	8	4					53				
Avaliar os métodos numéricos ponto de vista da eficiência computacional e da robustez .	4	3	1					12				
Usar o sistema computacional Matlab na implementação de alguns dos métodos estudados.			14					24				
Seleccionar rotinas do Matlab para resolver problemas propostos e discutir os resultados obtidos.			6					12				
Interpretar os erros de um resultado numérico em termos do condicionamento do problema e da estabilidade numérica do algoritmo usado.	3	2	1					10				
TOTAL	28	14	28					121			5	196

CURSO: **Ciências da Computação**

UNIDADE CURRICULAR: **Bases de Dados**

ÁREA CIENTÍFICA: **Tecnologias da Computação (TC)**

UC UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de **5** créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo	Trab° projecto		
	T	TP	PL	TC	S	OT	E					
Conhecer os principais conceitos relacionados com os Sistemas de Bases de Dados – Arquitectura, Administração e Exploração.	5							5				
Conhecer e aplicar os principais conceitos relacionados com o Modelo Relacional, Álgebra Relacional e Cálculo de Predicados a situações práticas correntes.	5	3	2					10	6			
Conhecer e aplicar a Linguagem SQL nas vertentes de Administração, Exploração e Privacidade de Dados em sistemas de bases de dados reais.	5	2	3					10	10			
Saber projectar e modelar bases de dados para sistemas de software.	8	5	5					10	14			
Conhecer os principais problemas relacionados com a concorrência, segurança e recuperação em Bases de Dados.	5							5				
Conhecer e manipular algumas das ferramentas para projecto, desenvolvimento e administração de Sistemas de Bases de Dados Relacionais.		4	4						10			
TOTAL	28	14	14					40	40		4	140

CURSO: Ciências da Computação

UNIDADE CURRICULAR: Computabilidade

ÁREA CIENTÍFICA: Ciências da Computação (CC)

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 6 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo	Trab° projecto		
	T	TP	PL	TC	S	OT	E					
Compreender os conceitos de Máquina de Turing e função parcial recursiva e descrever nestes modelos de computação soluções computacionais para problemas simples.	10	10						28				
Conhecer alguma da evidência para a Tese de Church-Turing e reconhecer a importância e as implicações desta.	3	3						10				
Aplicar funções universais em argumentos de computabilidade e explicar as principais ideias subjacentes à construção destas funções.	4	4						20				
Argumentar sobre a indecidibilidade de problemas em teoria da computabilidade.	4	4						20				
Relacionar os conceitos de decidibilidade e semi-decidibilidade de problemas.	3	3						10				
Compreender o conceito de problema NP e conhecer alguns exemplos deste tipo de problemas.	4	4						20				
TOTAL	28	28						108			4	168

CURSO: **Ciências da Computação**
 UNIDADE CURRICULAR: **Geometria**

Anexo 2

ÁREA CIENTÍFICA: **Matemática (M)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Distribuição das horas creditadas ao aluno para obtenção de **6** créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Laborato-riais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trabº grupo	Trabº projecto		
	T	TP	PL	TC	S	OT	E					
Capacidade de resolução de problemas de incidência e métricos.	8	8						32				
Reconhecimento de transformações geométricas básicas.	8	8						32				
Identificação de curvas e superfícies simples.	6	6						20				
Aquisição de conhecimentos básicos sobre geometria projectiva.	6	6						24				
TOTAL	28	28						108			4	168

CURSO: Ciências da Computação

UNIDADE CURRICULAR: Programação Concorrente

ÁREA CIENTÍFICA: Ciências da Computação (CC)

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 6 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo			Trab° projecto
	T	TP	PL	TC	S	OT	E					
Modelar sistemas concorrentes.	6		6					9				
Compreender os principais modelos e primitivas de programação concorrente em memória partilhada.	8		8					24				
Escrever aplicações concorrentes em memória partilhada.	4		4					18	10			
Compreender os principais modelos e primitivas de programação concorrente em sistemas distribuídos.	6		6					18				
Escrever aplicações concorrentes baseadas em passagem de mensagens.	4		4					19	10			
TOTAL	28		28					88	20		4	168

CURSO Ciências da Computação

UNIDADE CURRICULAR: **Computação Gráfica**ÁREA CIENTÍFICA : **Tecnologias da Computação (TC)**UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 6 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Laborato- riais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trabº grupo	Trabº projecto		
	T	TP	PL	TC	S	OT	E					
Compreender as Transformações Geométricas e os modelos de iluminação, incluindo a Matemática Subjacente.	8	4	4					16				
Compreender os processos matemáticos de definição de geometria (curvas, superfícies).	8	4	4					16				
Compreender o processo de animação via <i>key frames</i> .	4	1	1					6				
Compreender o modelo de eventos subjacente às aplicações gráficas.	4	2	2					6				
Compreender a história e o papel do VRML dentro da computação gráfica.	4							4				
Conceber e implementar sistemas gráficos.		3	3						60			
TOTAL	28	14	14					48	60		4	168

CURSO: **Ciências da Computação**

UNIDADE CURRICULAR: **Processos e Concorrência**

ÁREA CIENTÍFICA: **Ciências da Computação (CC)**

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de **6** créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo	Trab° projecto		
	T	TP	PL	TC	S	OT	E					
Compreender e comparar diferentes modelos e linguagens para especificar sistemas concorrentes.	10	4						30				
Modelar sistemas concorrentes de pequena e média complexidade usando redes de Petri.	6	8						10	20			
Especificar propriedades de segurança e animação usando lógica temporal.	4	4						10				
Compreender o mecanismo de verificação de modelos para a lógica temporal.	8	4						10				
Utilizar ferramentas de verificação de modelos para analisar propriedades de sistemas concorrentes.	0	8						8	20			
TOTAL	28	28						68	40		4	168

CURSO: Ciências da Computação

UNIDADE CURRICULAR: Semântica da Programação

ÁREA CIENTÍFICA: Ciências da Computação (CC)

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 5 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo			Trab° projecto
	T	TP	PL	TC	S	OT	E					
Conhecer as características das abordagens axiomática, operacional e denotacional à semântica das linguagens de programação.	4	4						6				
Atribuir significado formal a uma linguagem funcional simples (lambda-calculus).	6	6						9	10			
Utilizar sistemas de tipos para regular a boa formação de programas.	6	6						9	9			
Utilizar padrões de recursividade associados a tipos indutivos.	4	4						6				
Atribuir semântica à recursividade genérica – utilizar domínios semânticos contínuos.	8	8						12	19			
TOTAL	28	28						42	38		4	140

CURSO: Ciências da Computação

UNIDADE CURRICULAR: Teoria das Probabilidades

ÁREA CIENTÍFICA: Ciências da Computação (CC)

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 7 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente							Horas de trabalho independente			Horas de avaliação	Total
	Colectivas		Laborato- riais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trabº grupo	Trabº projecto		
	T	TP	PL	TC	S	OT	E					
Compreender os conceitos básicos de probabilidades.	8	5	4					10	8			
Conhecer e utilizar diversos modelos probabilísticos.	8	4	5					14	10			
Conhecer técnicas de simulação.	5	1	6					10	10			
Compreender e aplicar os vários tipos de convergências estocásticas.	7	4	5					12	10			
Adquirir a capacidade de explorar software para resolução de problemas.			8						18			
TOTAL	28	14	28					46	56	20	4	196

CURSO: Ciências da Computação

UNIDADE CURRICULAR: Teoria dos Números Computacional

ÁREA CIENTÍFICA: Matemática (M)

UC – ANUAL SEMESTRAL TRIMESTRAL OUTRA

OBRIGATÓRIA OPCIONAL

Anexo 2

Distribuição das horas creditadas ao aluno para obtenção de 6 créditos (ECTS)

Resultados de aprendizagem (RA)	Horas de contacto com o docente						Horas de trabalho independente			Horas de avaliação	Total	
	Colectivas		Labora- toriais	T. de campo	Seminário	Tutórias	Estágios	Estudo	Trab° grupo			Trab° projecto
	T	TP	PL	TC	S	OT	E					
Estudar alguns tipos de congruências não lineares.	6	8										
Aplicar os conhecimentos sobre fracções contínuas na resolução de certas equações.	6	8										
Conhecer os principais algoritmos de factorização.	8	6										
Conhecer os principais testes de primalidade e de pseudo-primalidade	8	6										
TOTAL	28	28						112			4	168