

# UNIDADE CURRICULAR 21090

## Programação

Docente(s): José Coelho

Ano Lectivo: 2013/2014

### PUC

#### O que é o PUC?

O PUC constitui um documento que visa orientar o processo de aprendizagem do estudante ao longo da Unidade Curricular a que se refere. É por isso um guião que requer uma leitura atenta e que lhe será útil ao longo de todo o percurso de aprendizagem. Aqui encontrará informação sobre as temáticas a estudar, as competências a desenvolver, informações sobre como se organiza o processo de aprendizagem, como utilizar e tirar partido do espaço virtual relativo a esta Unidade Curricular, o que se espera de si, como é avaliado, entre outros aspectos fundamentais para realizar da melhor forma este percurso.

## 1. A Unidade Curricular

### Apresentação da Unidade Curricular

Esta unidade curricular tem por objetivo o ensino de uma linguagem de programação estruturada imperativa, tendo sido escolhida para isso a linguagem C. O estudante ficará com bases sólidas sobre programação e os métodos de programação necessários para que possa abordar, mais tarde, linguagens de programação mais evoluídas e complexas.

## 2. Competências

### Competências a Desenvolver

Pretende-se que, no final desta Unidade Curricular, o estudante tenha adquirido as seguintes competências:

- Identificação dos conceitos da programação imperativa, de forma a utilizá-los em outras linguagens de programação;
- Produção de pequenos programas numa linguagem imperativa;
- Desenvolver algoritmos e estruturas de dados para pequenos problemas.

## 3. Roteiro

# Roteiro de conteúdos a trabalhar

Nesta Unidade Curricular serão trabalhadas as seguintes temáticas :

•	<b>Módulo 1</b> - Variáveis e estruturas de controlo	<ul style="list-style-type: none"><li>• 1 - Primeiro Programa</li><li>• 2 - Variáveis</li><li>• 3 - Condicionais</li><li>• 4 - Ciclos</li></ul>
•	<b>Módulo 2</b> - Funções, Vetores e Recursão	<ul style="list-style-type: none"><li>• 5 - Funções</li><li>• 6 - Mais Ciclos e Condicionais</li><li>• 7 - Vetores</li><li>• 8 - Procedimentos</li><li>• 9 - Recursão</li></ul>
•	<b>Módulo 3</b> - Memória, Estruturas e Ficheiros	<ul style="list-style-type: none"><li>• 10 - Memória</li><li>• 11 - Estruturas</li><li>• 12 - Ficheiros</li><li>• 13 - Truques</li></ul>

## 4. Metodologia

### Como vamos trabalhar?

A unidade curricular está dividida em três **módulos**. Os módulos vão abrindo ao longo do semestre, e em cada um deles existe a identificação dos **capítulos do livro** que têm de ser lidos, e um conjunto de **atividades formativas** que devem ser realizadas após a leitura dos capítulos. Cada atividade tem uma caixa de seleção, que o estudante pode selecionar após concluída a atividade, de modo a controlar mais facilmente o seu percurso, e por outro lado dar alguma informação à equipa docente da sua evolução.

Existem 3 atividades formativas pré-programação, sendo todas as restantes exercícios em que se pede para desenvolver um pequeno programa. A realização dos exercícios de pré-programação, destina-se a assegurar que os conceitos iniciais e essenciais à programação, são de facto compreendidos ao nível prático, e não apenas conceptual. As atividades formativas de programação estão classificadas em três cores, verde/azul/vermelha de forma a indicar o seu grau de dificuldade. A realização de todos os exercícios verdes e alguns azuis, é condição suficiente para que o estudante tenha a preparação necessária para obter aprovação na UC.

Para apoio no estudo, esclarecimento de dúvidas, bem como para cimentar a matéria, deve interagir com os colegas e equipa docente, através dos fóruns. Existem dois fóruns principais, o **fórum da turma**, e o **fórum de dúvidas**. O primeiro situa-se no espaço da turma, enquanto que o segundo situa-se no **espaço central**, comum a todas as turmas, e está dividido por módulos. O espaço da turma é o local está tudo o que diz respeito à avaliação, devendo a interação desenvolver-se mais no espaço central para assim tirar partido do maior número de participações.

No espaço central existe o **wiki de programação**, que funcionará como um quadro assíncrono aberto à edição tanto pela equipa docente como pelos estudantes. No wiki

estão respondidas as questões frequentes, e devem ser colocadas as explicações mais longas mesmo que pouco frequentes, além de outras utilizações como a publicação dos resultados globais de actividades letivas.

## Atividades Formativas (AFs)

As atividades formativas de pré-programação destinam-se a assegurar que os conceitos iniciais e essenciais à programação, estão consolidados a nível prático, e assim suavizar as naturais dificuldades de alguém que faz os seus primeiros programas. Nestas AFs a correção é automática, podendo o estudante ficar mais autónomo. No entanto não deve deixar de contactar a equipa docente e os colegas, face a dificuldades nestas ou em outras AFs.

As atividades formativas de programação são a maioria das atividades formativas, e essenciais à aquisição das competências. Programar aprende-se programando, e é exactamente isso que é pretendido com estas actividades formativas.

A realização individual de todas as AFs de programação verdes e algumas azuis, significa que o estudante está apto a obter uma **nota positiva** nas actividades de avaliação (e-fólios e p-fólio). O estudante que conseguir resolver todas as AFs de programação azuis, e algumas vermelhas, deverá conseguir uma **boa nota** nas actividades de avaliação, enquanto que o estudante que resolver todas as AFs de programação vermelhas, irá certamente obter aprovação com uma **excelente nota**.

Cada AF de programação tem uma pergunta associada. Essa pergunta deve ser respondida após a realização da AF no "**Registo de AFs realizadas**", obtendo-se assim uma chave para no "**Arquivo de AFs realizadas**" encontrar as **resoluções dos seus colegas** que entenderam partilhar o código, podendo assim comparar com a sua resolução. Pode também fazer o mesmo, e deve, no caso da sua resolução se afastar significativamente das resoluções já publicadas.

Veja apenas o código externo (professor e colegas), não o importe para o seu computador, utilizando-o apenas para comparar e melhorar o seu próprio código.

Após realizar uma AF, pode utilizar um utilitário que detecta parte de erros de qualidade de código (ver no wiki, "**Análise de AFs**"). Existe um critério de avaliação nos e-fólios relativo à qualidade do código, pelo que esta análise é importante de modo a não cometer erros de qualidade no e-fólio. Por outro lado, parte destes erros são cometidos devido à má compreensão de um conceito de programação, pelo que é essencial que veja os erros apontados e analise formas alternativas de não os cometer. Pode e deve também utilizar o resultado do utilitário para colocar questões no fórum, de modo a aprofundar as questões apontadas. Esta análise é mais relevante a partir do módulo 2.

Ao colocar a resposta no "**Registo de AFs realizadas**", a sua pontuação global da avaliação contínua será actualizada. É atribuído 1 ponto por cada AF verde realizada, 2 pontos por cada AF azul, e 4 pontos por cada AF vermelha. O *top 100* com os estudantes com mais pontos, irá surgir na barra da direita do espaço central.

Não faz sentido responder à pergunta de uma AF de programação por outra via que não seja a realização individual da AF (caso a pergunta possa ser respondida sem implementar código, pedir a um colega, utilizar uma aplicação para obter o resultado, obter o resultado manualmente, procurar na web, etc.). Caso não se sinta capaz de fazer uma AF, deve **pedir ajuda no fórum**.

Não faz sentido a realização de AFs em grupo, ou com a ajuda de um amigo ou explicador. A realização das AFs tem de ser um processo individual, verifique se está correta no "**Registo de AFs Realizadas**", e comunique com os seus colegas na UC. Caso as AFs sejam feitas com ajuda, ficará com AFs resolvidas mas sem o conhecimento para as resolver.

Não coloque o código de uma AF de programação no fórum para não comprometer a realização da AF dos seus colegas que ainda não a fizeram, ou no "**Arquivo de AFs realizadas**" sem estar codificado. Se o fizer pode desmotivar alguns dos seus colegas que ainda não fizeram a AF, que ao ver código que não compreendem e aparenta estar muito distante das suas capacidades, irão pensar em desistir, em vez de iniciar caminho, tentando resolver as AFs do nível em que estão. Tem hipótese de partilhar e ver código dos colegas no fórum em aberto, mas apenas o código relativo aos e-fólios, após a data de entrega, ou excepcionalmente, código de uma tentativa de resolução de uma AF, que não funcione e não consiga com as ajudas obtidas, colocar o código a funcionar.

Existem também três atividades formativas no módulo 2, com matéria não avaliada. Estas atividades formativas visam a introdução das linguagens de programação R (orientada para o cálculo científico), VBA (programação dentro de aplicações, utilizando-se o MS Excel) e Processing (orientada a artistas), com base nos programas do manual traduzidos de C. Cada estudante deve realizar estas atividades de acordo com o seu interesse, sendo no entanto aconselhado aos estudantes de Matemática a realização da AF da linguagem R, dado que é utilizada em outras UCs do curso.

## Interação

Existem dois espaços da unidade curricular de Programação, o espaço da turma, e o espaço central, comum a todas as turmas. Em cada espaço existe um fórum de comunicação entre estudantes e equipa docente, o **fórum de turma** e o **fórum de dúvidas**. O fórum de dúvidas do espaço central, encontra-se dividido por módulos. No espaço central existe ainda o **wiki de programação** que se destina a ser utilizado como se fosse um quadro assíncrono.

Pode gerir os fóruns da forma que entender, por exemplo:

- inscrever-se no fórum, recebendo **um email por cada mensagem** (o assunto inicia-se com o nome curto da UC, pelo que pode colocar um filtro no cliente de email, para agrupar todos os emails gerados por mensagens da UC);
- inscrever-se no fórum, mas configurar a conta para agrupar todas as mensagens do Moodle em **um só email diário** (editar o perfil);
- não se inscrever no fórum (ou anular a inscrição), **consultando regularmente o espaço da UC**.

A gestão dos fóruns é sua, tal como o nível de envolvimento na unidade curricular, nomeadamente na ajuda aos seus colegas no fórum, mas tenha em atenção que informações essenciais para a realização da unidade curricular são divulgadas no fórum.

Nos fóruns os estudantes devem procurar colocar **dúvidas o mais claro e contextualizadas quanto possível**, para evitar estar-se a responder à pergunta errada.

Os estudantes mais avançados devem procurar nos fóruns esclarecer os colegas com mais dificuldades. Este tipo de actividade tem vindo a ser considerado pelos estudantes mais colaborativos dos anos anteriores, como muito compensador, permitindo cimentar a matéria e reorganizar ideias. Uma ajuda pode ser simplesmente a indicação de uma página no livro, ou do wiki, onde o colega pode encontrar resposta à sua dúvida.

O **wiki de programação** deve ser utilizado para a colocação de esclarecimentos mais longos, constituindo assim uma forma de comunicação. O wiki tem tantos níveis hierárquicos quantos os que forem necessários, pelo que podem ser colocados exercícios adequados a uma determinada gama de estudantes, com resoluções passo-a-passo, bem como tudo o que for necessário, tal como um quadro numa sala de aula.

No wiki, ao contrário do quadro, não é necessário apagar nada, e funciona em modo assíncrono. Como o wiki não permite notificações por email, qualquer publicação de uma explicação no wiki, deve ter uma mensagem correspondente no fórum, a anunciar a explicação. Qualquer gralha, deve ser corrigida pela primeira pessoa a detectá-la, ou na dúvida, o mais seguro será colocar a questão no fórum, de forma a que a qualidade do texto no wiki seja mantida a um nível elevado.

Atendendo a que o wiki irá manter tanto uma estrutura de fácil leitura, como uma qualidade de texto elevada, é conveniente que antes de colocar uma dúvida no fórum, o estudante consulte o wiki para ver se não está já uma explicação no wiki que lhe sirva. Naturalmente que esta prática poderia ser estendida ao fórum, mas o fórum pode ter um tráfego muito elevado, o que inviabiliza a leitura total por parte de um estudante com menos disponibilidade. Não deve portanto ser levado a mal a colocação de uma dúvida repetida. No caso de repetição, deve-se preparar uma página no wiki de forma a dar uma resposta cabal e eficaz a uma dúvida comum a vários estudantes.

## 5. Recursos

# Bibliografia e outros recursos

### Bibliografia Obrigatória:

- Coelho, José - *Conceitos e exercícios de programação, utilizando Linguagem C*". Lisboa: Universidade Aberta, 2010 (Repositório Aberto: <http://hdl.handle.net/10400.2/1779>)
  - [Versão HTML, revisão 2012](#)
  - [Versão PDF, revisão 2012](#)

### Bibliografia Complementar:

- Fundamentos de Programação usando C, Marques de Sá, FCA – ISBN: 978-972-722-475-3
- Linguagem C, Luís Damas, FCA – ISBN: 978-972-722-156-1

A leitura da bibliografia complementar é opcional, e não dispensa a leitura da bibliografia obrigatória.

## 6. Avaliação

# Como vai ser a avaliação?

A avaliação assume o regime de avaliação contínua, não existindo a modalidade de exame final.

Os estudantes têm disponível um **Cartão de Aprendizagem**, pessoal, onde será creditada a avaliação que forem efectuando ao longo do semestre. O Cartão de Aprendizagem é um instrumento personalizado e cada estudante tem acesso apenas ao seu cartão.

## 6. Avaliação

### 6.1. Cartão de Aprendizagem

# Avaliação Contínua

Os estudantes creditam no seu Cartão de Aprendizagem – CAP – as classificações que obtiveram com a realização de vários trabalhos ao longo do semestre e a classificação obtida numa prova presencial, no final do semestre.

Ao longo do semestre ser-lhes-á proposto que elaborem pequenos trabalhos, designados **e-fólios**. A realização do conjunto dos e-fólios poderá levar à creditação do seu cartão (CAP) de um máximo de 8 valores. Nesta unidade curricular a avaliação compreende a realização de dois e-fólios, designados e-fólio A (cotação de 4 valores) e e-fólio B (cotação de 4 valores).

À classificação obtida nos **e-fólios** juntam-se os valores obtidos na prova presencial, designada **p-fólio**, tendo esta a duração máxima de 90min. Esta prova, *p-fólio*, terá a cotação global de 12 valores.

A classificação final na unidade curricular corresponde ao somatório das classificações obtidas em cada *e-fólio* e no *p-fólio*.

Um **e-fólio** é um pequeno documento digital elaborado de forma pessoal e incidindo sobre uma secção dos temas trabalhados.

O **p-fólio** consiste num documento escrito a realizar presencialmente, que complementa os *e-fólios* realizados electronicamente.

A aprovação na Unidade Curricular implica que o estudante obtenha um mínimo de 3.5 valores no conjunto dos *E-fólios* e um mínimo de 5.5 valores no *P-fólio*, e um valor mínimo de 9.5 valores no somatório dos *E-fólios* como *P-fólio*.

Na última semana lectiva, decorre um **e-fólio de recurso**, disponível para quem tenha reprovado nos e-fólios, substituindo o e-fólio com a pior cotação.

Nesta unidade curricular não existe a modalidade de exame.

## 6. Avaliação

### 6.2. Calendário de avaliação contínua

# Calendário de avaliação contínua

[O quadro seguinte deverá ser preenchido de acordo com o planeamento efectuado; ter em conta o número de e-fólios previstos e a valorização máxima de cada um ao preencher o quadro e as datas correspondentes ao semestre em questão.]

	Novembro	Dezembro	Janeiro	Fevereiro
<b><i>E-fólio A</i></b> [4 valores]				
Data da especificação do trabalho a realizar no <i>E-fólio A</i> e dos respetivos critérios de avaliação	22			
Envio do <i>E-fólio A</i> ao professor		2		
Indicação da classificação do <i>E-fólio A</i>		6		
<b><i>E-fólio B</i></b> [4 valores]				
Data da especificação do trabalho a realizar no <i>E-fólio B</i> e dos respetivos			3	

critérios de avaliação				
Envio do <i>E-fólio B</i> ao professor			13	
Indicação da classificação do <i>E-fólio B</i>			17	
<b><i>E-fólio C - Recurso</i></b>  [4 valores]				
Data da especificação do trabalho a realizar no <i>E-fólio C</i> e dos respetivos critérios de avaliação			17	
Envio do <i>E-fólio C</i> ao professor			27	
Indicação da classificação do <i>E-fólio C</i>			31	
<b><i>P-fólio</i></b> 12 valores				
Realização presencial				10 (tarde)*

\* - confirmar data e hora no site da universidade, uma semana antes da prova.

## 7. Plano de Trabalho

# Calendário e plano de trabalho

Este *Plano* apresenta a previsão da distribuição temporal das várias **Temáticas** de estudo, das atividades e respectivas orientações de trabalho, de modo a que possa planear,



organizar e desenvolver o seu estudo. Esta informação é complementada por orientações que deverá consultar com regularidade nesta *Sala de Aula Virtual*.

## 7. Plano de Trabalho

### 7.1. Primeiro mês

<b>Mês 1</b>		
	<b>Outubro</b>	<b>O que se espera do estudante</b>
	<b>1ª Semana</b> 7 de Outubro	<ul style="list-style-type: none"> <li>• Ler o <a href="#">Plano da Unidade Curricular</a> (PUC) e colocar as dúvidas que tiver no Fórum.</li> <li>• Ler os capítulos do módulo 1.</li> <li>• Instalar o compilador e seguir os exemplos do livro.</li> <li>• Consultar o wiki, colocar e seguir as dúvidas que vão surgindo, no fórum.</li> </ul>
	<b>2ª Semana</b> 14 de Outubro	<ul style="list-style-type: none"> <li>• Fazer todos os exercícios pré-programação</li> <li>• Fazer todos os exercícios verdes (AF1)</li> <li>• Fazer alguns exercícios azuis (AF2)</li> <li>• Caso tenha feito todos os exercícios azuis, avançar para os vermelhos (AF3)</li> <li>• Consultar o wiki, colocar e seguir as dúvidas que vão surgindo, no fórum.</li> </ul>
	<b>3ª Semana</b> 22 de Outubro	<ul style="list-style-type: none"> <li>• Semana de análise. Escolha dos exercícios que fez os mais complexos, analise o código com o utilitário de análise (ver Análise de AFs) e compare com as resoluções dos seus colegas.</li> <li>• Leia novamente os capítulos do módulo 1, com especial atenção na lista de erros possíveis, no final de cada capítulo.</li> <li>• Consultar o wiki, colocar e seguir as dúvidas que vão surgindo, no fórum.</li> </ul>
	<b>4ª Semana</b> 28 de Outubro	<ul style="list-style-type: none"> <li>• Começar a ler os capítulos do módulo 2.</li> <li>• Especial atenção deve ser dada para o capítulo Funções, o qual</li> </ul>

deve ser entendido completamente, incluindo as execuções passo-a-passo.

- Consultar o wiki, colocar e seguir as dúvidas que vão surgindo, no fórum.

## 7. Plano de Trabalho

### 7.2. Segundo mês

Mês 2		
	Novembro	O que se espera do estudante
	<b>1ª Semana</b> 4 de Novembro	<ul style="list-style-type: none"><li>• Acabar de ler os capítulos do módulo 2</li><li>• Fazer alguns exercícios verdes (AF4)</li><li>• Consultar o wiki, colocar e seguir as dúvidas que vão surgindo, no fórum.</li></ul>
	<b>2ª Semana</b> 11 de Novembro	<ul style="list-style-type: none"><li>• Fazer os restantes exercícios verdes (AF4)</li><li>• Fazer alguns exercícios azuis (AF5)</li><li>• Caso tenha feito todos os exercícios azuis, avançar para os exercícios vermelhos (AF6)</li><li>• Veja as AFs extra, sobre as linguagem de programação R, VBA e Processing.</li><li>• Consultar o wiki, colocar e seguir as dúvidas que vão surgindo, no fórum.</li></ul>
	<b>3ª Semana</b> 18 de Novembro	<ul style="list-style-type: none"><li>• Semana de análise. Escolha dos exercícios que fez os mais complexos, analise o código com o utilitário de análise (ver Análise de AFs) e compare com as resoluções dos seus colegas.</li><li>• Leia novamente os capítulos do módulo 2, com especial atenção na lista de erros possíveis, no final de cada capítulo.</li><li>• Consultar o wiki, colocar e seguir as dúvidas que vão surgindo, no fórum.</li><li>• No final da semana será divulgado o enunciado do e-fólio A. Faça uma primeira leitura e coloque as suas dúvidas sobre o enunciado o mais cedo possível.</li></ul>

	<b>4ª Semana</b> 25 de Novembro	<ul style="list-style-type: none"> <li>• Semana de realização do e-fólio A (disponível na 6ª-feira anterior, até à 2ª-feira da semana seguinte).</li> <li>• Utilize os primeiros dias da semana para realizar o e-fólio, e procure esclarecer eventuais ambiguidades no enunciado.</li> <li>• Pode colocar dúvidas da matéria, mas apenas se estiverem convenientemente distanciadas do e-fólio em curso.</li> </ul>
--	------------------------------------	--

## 7. Plano de Trabalho

### 7.3. Terceiro mês

<b>Mês 3</b>		
	Dezembro	O que se espera do estudante
	<b>1ª Semana</b> 2 de Dezembro	<ul style="list-style-type: none"> <li>• Análise do e-fólio A. Partilhe o seu código, e veja o dos seus colegas.</li> <li>• Começar a ler os capítulos do módulo 3.</li> <li>• Consultar o wiki, colocar e seguir as dúvidas que vão surgindo, no fórum.</li> </ul>
	<b>2ª Semana</b> 9 de Dezembro	<ul style="list-style-type: none"> <li>• Acabar de ler os capítulos do módulo 3</li> <li>• Fazer alguns exercícios verdes (AF7)</li> <li>• Consultar o wiki, colocar e seguir as dúvidas que vão surgindo, no fórum.</li> </ul>
	<b>3ª Semana</b> 16 de Dezembro	<ul style="list-style-type: none"> <li>• Fazer os restantes exercícios verdes (AF7)</li> <li>• Fazer alguns exercícios azuis (AF8)</li> <li>• Caso tenha feito todos os exercícios azuis, avançar para os exercícios vermelhos (AF9)</li> <li>• Consultar o wiki, colocar e seguir as dúvidas que vão surgindo, no fórum.</li> </ul>
	<b>4ª Semana</b> 23 de Dezembro	<ul style="list-style-type: none"> <li>• Semana de paragem do Natal e Ano Novo</li> </ul>

## 7. Plano de Trabalho

### 7.4. Quarto mês

Mês 4		
	Janeiro	O que se espera do estudante
	<b>1ª Semana</b> 30 de Dezembro	<ul style="list-style-type: none"><li>• Semana de análise. Escolha dos exercícios que fez os mais complexos, analise o código com o utilitário de análise (ver Análise de AFs) e compare com as resoluções dos seus colegas.</li><li>• Leia novamente os capítulos do módulo 3, com especial atenção na lista de erros possíveis, no final de cada capítulo.</li><li>• Consultar o wiki, colocar e seguir as dúvidas que vão surgindo, no fórum.</li></ul>
	<b>2ª Semana</b> 6 de Janeiro	<ul style="list-style-type: none"><li>• Semana de realização do e-fólio B (disponível na 6ª-feira anterior, até à 2ª-feira da semana seguinte).</li><li>• Utilize os primeiros dias da semana para realizar o e-fólio, e procure esclarecer eventuais ambiguidades no enunciado.</li><li>• Pode colocar dúvidas da matéria, mas apenas se estiverem convenientemente distanciadas do e-fólio em curso.</li></ul>
	<b>3ª Semana</b> 13 de Janeiro	<ul style="list-style-type: none"><li>• Análise do e-fólio B. Partilhe o seu código e veja o dos seus colegas.</li><li>• Consultar o wiki, colocar e seguir as dúvidas que vão surgindo, no fórum.</li></ul>
	<b>4ª Semana</b> 20 de Janeiro	<ul style="list-style-type: none"><li>• Semana de realização do e-fólio de Recurso (disponível na 6ª-feira anterior, até à 2ª-feira da semana seguinte). .</li><li>• Se reprovou aos e-fólios, faça nesta semana o e-fólio de Recurso, caso contrário utilize esta semana para revisão da matéria.</li></ul>

- |  |  |   |
|--|--|---|
|  |  | <ul style="list-style-type: none"><li>• Reveja os exercícios de programação que não conseguiu fazer, faça-os agora.</li><li>• Consultar o wiki, colocar e seguir as dúvidas que vão surgindo, no fórum.</li></ul> |
|--|--|---|