 IPG Politécnico da Guarda Escola Superior de Tecnologia e Gestão	GUIA DE FUNCIONAMENTO DA UNIDADE CURRICULAR		MODELO PED.008.02

Curso	Engenharia Informática			Ano letivo	2015-2016		
Unidade Curricular	Arquitectura de Computadores			ECTS	6		
Regime	Obrigatório						
Ano	2º	Semestre	2ºsem	Horas de trabalho globais			
Docente (s)	Luis Figueiredo			Total	168	Contacto	75
Coordenador	António Martins						

GFUC previsto

1. OBJETIVOS DE APRENDIZAGEM

Após a conclusão da UC, os alunos deverão ser capazes de:

1. Perspetivar a evolução futura dos computadores com base no conhecimento da sua evolução até aos dias de hoje.
2. Descrever as operações básicas matemáticas e lógicas e diferentes formas de representação de quantidades numéricas e não numéricas.
3. Identificar a arquitectura geral dos computadores.
4. Desenvolver algoritmos e implementar programas e/ou procedimentos em linguagens de baixo nível e comparar o seu desempenho com programas e/ou procedimentos implementados em linguagens de alto nível.
5. Identificar diferentes técnicas de aumento da velocidade de processamento das aplicações, quer ao nível do hardware quer ao nível do software.
6. Utilizar microcontroladores para aquisição e controlo de sinais analógicos e digitais bem como efetuar a comunicação do microcontrolador com um computador.

2. CONTEÚDOS PROGRAMÁTICOS

1. Breve história dos computadores
2. Revisão de conceitos fundamentais para arquitectura de computadores
 - a. Bits, Bytes, Words

- b. Representação de números em diferentes bases
 - c. Operações com números em diferentes bases
 - d. Representação de números negativos
 - e. Floating points
 - f. Representação de dados não numéricos
3. Arquitectura geral dos computadores
- a. Organização geral de um CPU
 - b. Diferentes tipos de BUS
 - c. Dispositivos de I/O
4. Arquitectura 80XXX.
- a. Arquitecturas CISC/RISC
5. Organização e funcionamento da memória
- a. Diferentes hierarquias de memória
 - b. Modo real/modo protegido
 - c. Memória virtual
 - d. Memória cache
 - e. Alinhamento da memória e a sua influência no desempenho
6. Introdução à programação em baixo nível
- a. Instruções
 - b. Integração de Assembly com linguagens de alto nível
 - c. Medição do desempenho de aplicações
7. Técnicas de aumento da velocidade de processamento
- a. Pipeline
 - b. SIMD
 - c. Arquitectura superescalar
 - d. Previsão de saltos
 - e. Execução especulativa
8. Introdução aos microcontroladores
- a. Descrição do Arduino ao nível do hardware e software
 - b. Desenvolvimento de aplicações de leitura de sinais analógicos e digitais

- c. Desenvolvimento de aplicações de controlo de sinais analógicos e digitais.
- d. Desenvolvimento de aplicações de comunicação com um computador
- e. Utilização de interrupts: vantagens e limitações

3. DEMONSTRAÇÃO DA COERÊNCIA DOS CONTEÚDOS PROGRAMÁTICOS COM OS OBJETIVOS DA UC

Os conteúdos 1, 3 e 4 permitem atingir o objetivo 1 da UC, uma vez que se mostra a evolução da arquitetura dos computadores, não só de uma forma genérica como também de uma forma específica.

O objectivo 2 é atingido pelo conteúdo 2 sendo um pilar essencial para atingir os objectivos seguintes.

O objetivo 3 é atingido pela conjugação de todos os conteúdos do programa uma vez que cada uma delas tem uma contribuição para o conhecimento da arquitectura dos computadores.

O objetivo 4 é atingido pelos conteúdos 4, 5 e 6 uma vez que para desenvolver aplicações de baixo nível é necessário conhecer a arquitetura, o modelo de memória, e a linguagem de programação.

O objectivo 5 é atingido pelos conteúdos 4, 5, 6 e 7 uma vez que os mesmos enfocam as diferentes técnicas referidas nesse objectivo.

O objetivo 6 é atingido pelo conteúdo 8 que introduz os alunos, com aplicações práticas, no mundo dos microcontroladores.

4. BIBLIOGRAFIA PRINCIPAL

Obrigatória

Apontamentos do docente

IPG Politécnico da Guarda Escola Superior de Tecnologia e Gestão	GUIA DE FUNCIONAMENTO DA UNIDADE CURRICULAR	MODELO PED-008.02
--	--	-------------------

com
hand

John L. Hennessy, David A. Patterson. Computer Architecture: A Quantitative Approach, 2003

<http://spike.scu.edu.au/~barry/interrupts.html>

Recomendada

<http://www.intel.com/content/www/us/en/processors/architectures-software-developer-manuals.html>

<http://www.arduino.cc/>

5. METODOLOGIAS DE ENSINO (REGRAS DE AVALIAÇÃO)

Metodologias de ensino:

1. Lição expositiva
2. Lição interativa
3. Resolução de problemas
4. Trabalho de projeto

Regras de avaliação:

Avaliação contínua:

1. 7,5% para a presença em TODAS as aulas. Só será considerado para quem não der qualquer falta durante o semestre lectivo.
2. 7,5% para a colocação de cinco dúvidas, em 5 semanas diferentes antes da penúltima semana de aulas, no gabinete do docente ou através de correio electrónico.
3. 15% para o conjunto de 4 pequenos trabalhos práticos propostos ao longo do ano. Só serão considerados os trabalhos entregues nas datas definidas.
4. 35% para o trabalho final proposto. A nota mínima é de oito valores.

Luna

IPG Politécnico da Guarda Escola Superior de Tecnologia e Gestão	GUIA DE FUNCIONAMENTO DA UNIDADE CURRICULAR	MODELO PED.008.02
--	--	----------------------

dank

5. 35% para a frequência a realizar no final do semestre. A nota mínima é de oito valores.

Avaliação por exame final na Época Normal:


1. 15% relativos aos 4 trabalhos feitos ao longo do ano
2. 35% para o trabalho final proposto. A nota mínima é de oito valores.
3. 50% para o exame teórico. A nota mínima é de oito valores.

Avaliação por exame final na Época de Recurso ou Época Especial:

1. 50% para o trabalho final proposto. A nota mínima é de oito valores.
2. 50% para o exame teórico. A nota mínima é de oito valores.

6. DEMONSTRAÇÃO DA COERÊNCIA DAS METODOLOGIAS DE ENSINO COM OS OBJETIVOS DA UNIDADE CURRICULAR

1. Lição expositiva está coerente com os objetivos devido à necessidade de apresentar os conteúdos teóricos aos alunos, para que depois possam começar a desenvolver os trabalhos práticos da disciplina. Enquadram-se especialmente neste ponto os objetivos 1, 3 e 5.
2. Lição interativa está coerente com os objetivos pois a demonstração prática de uma determinada solução potencia a procura de soluções para novos problemas. Serão particularmente abrangidos por este tipo de lição os pontos 2 e 4.
3. Resolução de problemas está coerente com os objetivos uma vez que é absolutamente fundamental os alunos fazerem os seus próprios programas para não só consolidarem os conhecimentos teóricos apreendidos, como também para ganharem experiência no saber fazer e não apenas no saber como se faz. Os pontos 4 e 8 dos objetivos têm aqui a sua principal forma de ser atingido.

 <p data-bbox="518 179 726 302"> Politécnico da Guarda Escola Superior de Tecnologia e Gestão </p>	<p data-bbox="766 201 1236 280" style="text-align: center;">GUIA DE FUNCIONAMENTO DA UNIDADE CURRICULAR</p>	<p data-bbox="1284 257 1404 313" style="text-align: center;"> MODELO PED-008-02 </p>
---	--	---

4. Trabalho de projeto está coerente com os objetivos pois abrange o desenvolvimento de uma aplicação que permite aos alunos exercitar as suas capacidades de concepção e realização prática, o que implica o desenvolvimento de algoritmos e a implementação de programas e/ou procedimentos tal como conta do objetivo 4.

7. REGIME DE ASSIDUIDADE

Definido nas regras de avaliação

8. CONTATOS E HORÁRIO DE ATENDIMENTO

Gabinete 9, mail luis.figueiredo@ipg.pt


9. OUTROS

TRABALHOS OBRIGATÓRIOS AO LONGO DO ANO LECTIVO

1. Elaborar uma rotina em assembly para escrever os valores sequenciais de 0 a 255 num array de bytes com 100.000 elementos.
2. Elaborar uma rotina em assembly otimizada para escrever os valores sequenciais de 0 a 255 num array de bytes com 100.000 elementos, e comparar os tempos de execução com uma rotina feita em C.
3. Elaborar um programa para o Arduino que permita acender sequencialmente um de 8 leds. O tempo que cada led fica aceso será, no mínimo, de 30ms e, no máximo, de 3s sendo controlado por uma resistência variável. A direção do movimento dos leds deverá ser controlada por um interruptor.
4. Elaborar um programa para o Arduino que utilize interrupts para ler o valor de um sonar. O valor lido deverá ser enviado para o computador.

TRABALHOS FINAIS PROPOSTOS

1. Elaborar um programa que envie dados através da porta série e outro que os receba.
2. Elaborar uma rotina que utilize as instruções SIMD para somar valores inteiros de dois vectores.

 IPG Politécnico da Guarda Escola Superior de Tecnologia e Gestão	GUIA DE FUNCIONAMENTO DA UNIDADE CURRICULAR	MODELO PED.008.02
--	--	-------------------

lun
ant

3. Elaborar uma rotina que utilize as instruções SIMD para implementar um produtório de valores inteiros de dois vectores.
4. Elaborar uma rotina que não utilizando condições permita efectuar, de uma forma eficiente, o seguinte código:


```
void binariza(unsigned char *pt, unsigned char limite, int len)
{
    int x;
    for(x=0;x<len;++x)
        if(pt[x]<limite)
            pt[x]=0;
        else
            pt[x]=255;
}
```
5. Implementar uma rotina que transfira o mais rapidamente possível memória entre duas zonas.
6. Elaborar um programa para o Arduino que implemente uma estação meteorológica e faça a publicação dos dados na internet.
7. Elaborar um programa para o Arduino que implemente um mecanismo de controlo da velocidade de um motor.
8. Aceitam-se propostas de alunos para a realização de outros trabalhos.

Regras Gerais

Os trabalhos serão feitos em grupos de dois, não sendo permitido, sob qualquer pretexto, a formação de grupos maiores. Apesar do trabalho ser em grupo a avaliação será individual, o que significa que um dos elementos poderá ter positiva e o outro negativa.

Os relatórios dos trabalhos efectuados ao longo do semestre deverão ser entregues na aula prática seguinte àquela em que foram realizados ou iniciados. Só poderá entregar esse relatório quem tenha estado na respectiva aula prática.

O trabalho final obrigatório será escolhido pelos alunos de entre a lista proposta ou, em casos excepcionais, e com o acordo do docente, poderá ser um outro proposto pelos alunos dando-se preferência a trabalhos que utilizem o Arduino. O relatório deste trabalho deverá ser entregue 2 dias úteis antes da data da frequência. A sua apresentação e defesa será feita no dia da frequência tendo cada grupo 10 minutos para fazer essa apresentação.

Todas as situações detectadas de cópia, mesmo que parcial, de relatórios e/ou trabalhos entre grupos implicará a anulação dos respectivos trabalhos. A mesma situação se verificará se a cópia tiver como origem trabalhos de anos anteriores.

Chama-se a atenção para as questões de plágio de livros ou de informações na Internet. Sempre que seja necessário fazer uma transcrição ela deverá ser feita em *“itálico”*, com aspas, e com uma referência clara para a origem do texto.

Data: 22 de fevereiro de 2016

O regente da UC



(Luis Figueiredo)

O coordenador da área



(António Martins)