Ficha de Disciplina - (CPRO)

(Complementos de Programação)

(Engenharia Electrotécnica - Electrónica e Computadores)

Ano Lectivo: 2006/2007

Grupo de disciplinas: Engenharia da Programação

	Semestral	**Obrigatória**		
Regime	Diurno	**Horas/semana**	**T**-Teórica	2
Ano	4º		**TP**-Teórico-Prática	2
Semestre	1º		**PL**-Prática-Laboratorial	2

ECTS 7

Enquadramento da disciplina

A disciplina de Complementos de Programação faz parte do 4º ano - 1ºsemestre do 2º ciclo do curso de Engenharia Electrotécnica – Electrónica e Computadores ramo Automação e Sistemas. Esta disciplina visa dar conhecimentos estruturantes na área científica dos Sistemas de Computação.
Requer conhecimentos prévios específicos ao nível de:
- algoritmia e programação em C.
- arquitectura de computadores
- lingua inglesa (leitura e compreensão)

Objectivos

Objectivos Gerais:

O aluno deverá ser capaz de:
• Analisar e adaptar programas existentes e elaborar novos programas codificados em C.

• Compreender e assimilar conhecimentos avançados de sistemas operativos multi-processo e multi-utilizador. Estudo do modelo UNIX.

• Compreender os conceitos fundamentais da programação multi-processo e aplicá-los em exemplos concretos utilizando a linguagem C e o sistema operativo Linux.

• Utilizar o sistema operativo Linux, ferramentas de gestão de projecto (compilação e debugger) e mecanismos de cooperação entre tarefas.

• Compreender os conceitos da programação orientada a objectos e sua implementação na linguagem C++.

• Assumir atitudes de aprendizagem activa, colaborativa e responsável, de trabalho persistente e de aplicação de espírito crítico na análise e resolução de problemas.

Objectivos Específicos (Competências) :
1 - Intermédios
O aluno deverá ser capaz de:
•Mediante a apresentação de um algoritmo enunciar a sua funcionalidade e adaptá-lo a novas exigências.

• Mediante a apresentação de um problema descrever o respectivo algoritmo considerando os conceitos da programação multi-processo através da linguagem C e o sistema operativo Linux.

• Mediante a apresentação de programas explorar a respectiva funcionalidade no sistema operativo Linux,recorrendo a ferramentas de gestão de projecto (compilação e debugger) e mecanismos de cooperação entre tarefas.

• Mediante a apresentação de programas caracterizar cada uma das instruções e a funcionalidade global do programa.

• Mediante a apresentação de uma tarefa descrever o respectivo algoritmo e sua implementação na linguagem C++.

2 - Terminais
O aluno deverá ser capaz de:

•Mediante a apresentação de um problema, considerando os conceitos da programação multi-processo, conceber o respectivo algoritmo e elaborar o programa codificado em C no ambiente LINUX.

•Mediante a apresentação de um programa usar ferramentas de gestão de projecto (compilação e debugger) e mecanismos de cooperação entre tarefas para a sua execução.

•Recolher, seleccionar e interpretar informação relevante, a partir da Web, para a disciplina e curso.

•Integrar os conhecimentos, desenvolver soluções e emitir juízos de valor na realização e apresentação/discussão dos trabalhos.

Estratégias metodológicas (métodos e técnicas pedagógicas) e actividades de treino

Nas aulas teóricas serão utilizados o método expositivo e interrogativo e sempre que adequado serão também utilizadas diversas técnicas do método activo (tempestade de ideias, estudo de casos) .

Nas aulas teórico-práticas e práticas laboratoriais serão utilizados preferencialmente técnicas do método activo como trabalho de grupo, estudo de casos e aprendizagem baseada em problemas.

Na globalidade as estratégias usadas visam ajudar o aluno a assumir uma atitude de aprendizagem activa, colaborativa e responsável, trabalho persistente e de aplicação de espírito crítico na análise e resolução de problemas.

Programa

1. Introdução aos sistemas operativos
1.1 Noções básicas de arquitectura de computadores.
1.2 História dos sistemas operativos: gerações de sistemas operativos
1.3 Estrutura genérica de um sistema operativo. Funções e características de um sistema operativo.
1.4 Tipos de sistemas operativos e evolução dos sistemas operativos
1.5 O modelo UNIX

2. Processos
2.1 Gestão de processos
2.2 Signals
2.3 Sincronização

3. Comunicação entre processos
3.1 Pipes
3.2 Memória partilhada
3.3 Filas de mensagens
3.4 Sockets

4. Introdução à Programação Orientada a Objectos
4.1 Conceitos
4.2 A linguagem C++

Bibliografia

"Advanced Linux Programming", Mitchell, Samuel, New Riders Publishing, 2001. (http://www.advancedlinuxprogramming.com/)

"Unix for Programmers and Users, Glass, Prentice Hall, 1993.

"Interprocess Communications in Linux", The Nooks & Crannies, Prentice Hall, 2003. (http://e-books.amagrammer.net/Linux/)

"The C Programming Language", B. W. Kerningham, D. M. Ritchie, Prentice Hall Software Series.

"C: How to Program" – H. Deitel, P. Deitel – Prentice Hall.

"Elementos de Programação em C", Pedro Guerreiro, FCA.

"Fundamental de Linux" - Paulo Trezentos/António Cardoso, FCA 2001.

"Programação em C++, Algoritmos e Estruturas de Dados", Pimenta Rodrigues, Pedro Pereira e Manuela Sousa, FCA, 2000.

"Tópicos das Aulas Teóricas", Complementos de Programação, Ana Maria Madureira, 2006.

Técnicas, instrumentos e critérios de avaliação

Avaliação durante o Período Lectivo:
A avaliação sumativa durante o período lectivo (nota de frequência) é efectuada pela realização de um trabalho de programação, extra aulas, com um peso de 30% na classificação final. O trabalho será realizado em grupo mas apresentado e discutido individualmente. O trabalho de programação é obrigatório. Os alunos (qualquer que seja o seu regime) que não entreguem o trabalho não têm nota de frequência (NF).

O trabalho prático deve ser entregue até ao exame da disciplina na época normal em data a marcar pelos respectivos professores, mas para ser considerado, a evolução do trabalho será acompanhada nas aulas práticas durante o período lectivo.

Os alunos trabalhadores estudantes que pedirem isenção de avaliação de frequência a esta disciplina não estão dispensados do trabalho prático.

Avaliação na Prova de Exame:
A nota obtida pelo aluno na avaliação por prova de exame corresponde a 70% da sua classificação final.

A nota mínima da prova de exame é de 8 valores.

Classificação Final da Disciplina:
NotaFinal= (xNFREQ + yPE)/(x + y)
x = 30% Min NFREQ = 8 val
y = 70% Min PE = 8 val

Melhoria de Nota de Avaliação:
A melhoria de nota da disciplina é feita de forma semelhante ao efectuado para a avaliação normal da disciplina. O exame tem um peso idêntico ao que existiu quando o aluno obteve aprovação na disciplina.

Tipo de manutenção de nota de frequência

Permite manter a nota de frequência por um ano lectivo

Ano/Período Lectivo de obtenção da nota de frequência

–

Nota biográfica do regente da cadeira

A docente responsável pela disciplina é doutorada em Engenharia de Produção e Sistemas – Inteligência Artificial e Escalonamento Distribuído pela Universidade do Minho e exerce as funções de equiparada a Professor Adjunto no Departamento de Engenharia Informática.

Observações

	Nome
Director de Curso:	José Antonio Tenreiro Machado
Orientador do grupo de disciplinas:	Maria da Conceição Carvalho Benta de Oliveira Neves
Responsável pela disciplina:	Ana Maria Dias Madureira Pereira
Docentes que leccionam a disciplina:	Ana Maria Dias Madureira Pereira Antonio Manuel de Sousa Barros Nuno Miguel Vieira Morgado