



PROGRAMA GERAL DO COMPONENTE CURRICULAR- PGCC¹

I IDENTIFICAÇÃO DO COMPONENTE CURRICULAR

1.1 Natureza do componente: (X)Disciplina ()Atividades da prática² ()Estágio Supervisionado Obrigatório
()Trabalho de Conclusão de Curso – TCC
1.2 Nome do componente: PROGRAMAÇÃO CONCORRENTE E DISTRIBUÍDA (Optativa)
CÓDIGO: 0805041-1 **CRÉDITOS:** 04 **CARGA HORÁRIA:** 60 h.a.
Pré-Requisito: **Código:**
Curso: Ciência da Computação **Período:** **Turno:** Matutino **Ano/Semestre:** 2012.2
Professor (a): Sebastião Emídio Alves Filho

II EMENTA

Conceitos Básicos em Programação Distribuída: concorrência, distribuição, paralelismo, sistema distribuído, aplicações distribuídas, paralelismo X distribuição, bibliotecas do sistema operacional, requisitos para suporte a programação distribuída, comunicação entre processos, modelos de estruturação de programas distribuídos: Threads, Sockets, RPC - Chamada Remota de Procedimento e CORBA – Common Object Broker Architecture.

III OBJETIVOS

Introduzir conceitos básicos e avançados sobre as principais metodologias para programação concorrente e distribuída: comunicação entre processos, threads, sockets, objetos distribuídos e serviços. Fazer com que os discentes entendam e implementem exemplos de programas em clusters, grids e nuvem.

IV CONTEÚDO

1a Unidade

1. Conceitos básicos

- 1.1. Taxonomia de Flynn – sistemas multiprocessadores x sistemas multicomputadores
- 1.2. Conceitos e características de processos
- 1.3. Comunicação entre processos
- 1.4. Processos em C em sistemas UNIX-Like

2. Threads

- 2.1. Conceito de threads – diferença threads x processos
- 2.2. Sincronização entre Threads
- 2.3. Problemas clássicos de programação concorrente
- 2.4. Pacotes para controle e sincronização de threads em Java

2a Unidade

3. Sockets e objetos distribuídos

- 3.1. Conceito de sockets – modelo cliente/servidor
- 3.2. Sockets com TCP
- 3.3. Sockets (datagrama) com UDP
- 3.4. URL's e conexões
- 3.5. Sockets em java

4. Objetos distribuídos

- 4.1. Conceitos de RPC e objetos distribuídos
- 4.2. Funcionamento das bibliotecas de objetos distribuídos: CORBA e RMI
- 4.3. RMI em Java
- 4.4. Transferência e ativação de objetos remotos

3a Unidade

5. Orientação a serviços

- 5.1. Conceitos de orientação a serviços
- 5.2. WebServices: criação, localização e instanciação
- 5.3. Composição de serviços
- 5.4. Orquestração de serviços
- 5.5. Padrões de projetos para programação distribuída

6. Clusters, Grids e Nuvem

- 6.1. Conceito e taxonomia
- 6.2. Aplicações e simuladores
- 6.3. Implantação de ambientes

V METODOLOGIA

- Aulas expositivas em sala
- Aulas práticas em laboratório
- Utilização de simuladores
- Utilização do AVA Moodle

VI PROCEDIMENTOS DE AVALIAÇÃO DA APRENDIZAGEM

- 1ª unidade – Prova escrita, lista de exercícios práticos
- 2ª unidade – Prova escrita, lista de exercício práticos
- 3ª unidade – Prova escrita, seminários, projeto prático final da disciplina

VII REFERÊNCIAS

Bibliografia básica

ANDREWS, Gregory. Foundations of Multithreaded, Parallel and Distributed Programming -Addison- Wesley, 1999.

BAL, Henri. Programming Distributed Systems - Prentice Hall, 1990.

HORSTMANN, Cay s. and CORNELL, Gary . Core Java 2 – Volume II – Advanced Features, Prentice Hall, 1998

Bibliografia Complementar

GRAMA, Ananth et al. Introduction TO Parallel Computing. 2. ed. Harlow: Pearson Addison Wesley, 2003. p. 636. ISBN 0-201-64865-2.

VIII OUTRAS OBSERVAÇÕES

Aprovado pela Comissão do PPC em ____/____/____

Professor(a)

Presidente da Comissão do PPC