

INE 5645 – PROGRAMAÇÃO PARALELA E DISTRIBUÍDA

Identificação:

Disciplina: INE5645 - Programação Paralela e Distribuída

Carga horária: 72 horas-aula Teóricas: 36 Práticas: 36

Período: 2º semestre de 2014 (em processo de aprovação)

Curso:

Sistemas de Informação (238)

Requisito(s):

INE5611 - Sistemas Operacionais

Ementa:

Modelos de interação entre processos. Compartilhamento de memória e mecanismos de sincronização. Troca de mensagens e mecanismos de comunicação. Problemas clássicos. Princípios de implementação. Programação em redes de computadores. Programação distribuída. Linguagens paralelas e distribuídas.

Objetivo(s):

Geral: Apresentar conceitos e técnicas de programação paralela e distribuída.

Apresentar conceitos e técnicas de programação paralela e distribuída.

Específicos:

Apresentar e exercitar a programação paralela com sincronização e troca de mensagens.

Apresentar exemplos de interfaces de programação e linguagens paralelas e distribuídas

Conteúdo Programático:

Introdução à Programação Paralela e Distribuída [4 horas-aula]

Vantagens e Dificuldades

Plataformas de Execução

Suporte Computacional

Programação Paralela [16 horas-aula]

Processos

Threads

Paralelismo em Java

Controle de Concorrência [18 horas-aula]

Monitores

Locks

Semáforos

Concorrência na API Java

Programação Distribuída [8 horas-aula]

Modelos de sistemas distribuídos

Elementos básicos da comunicação

Comunicação por passagem de mensagem

Comunicação entre Processos [26 horas-aula]

Pipes

Sockets

RMI

Metodologia:

AT (Aula Teórica), AP (Aula Prática), TP (Tarefa Prática), TT (Tarefa Teórica)

Unidade I - Introdução à Programação Paralela e Distribuída (AT)

Unidade II - Programação Paralela (AT, AP, TP) (Concorrência)

Unidade III - Monitor, Locks e Semáforos (AT, AP, TP)

IV - Programação Distribuída com Sockets (AT, AP, TP)

Unidade V - Programação Distribuída com RMI (AT, AP, TP)

Unidade VI - Eventos e Notificações - Sockets ou RMI (AT, AP, TP)

Avaliação:

Os alunos serão avaliados com base em seu desempenho nas seguintes atividades:

2 Provas teóricas (P1 e P2);

7 Atividades de laboratório (LAB1, LAB2, LAB3, LAB4(a,b,c,d), LAB5, LAB6);

LAB6 é feita na forma de um projeto.

A média das provas (MP) será calculada aplicando a fórmula:

$$MP = (P1 + P2)/2.$$

A média das tarefas práticas (MT) será atribuída aplicando a fórmula

$$MT = [0,15*(LAB1) + 0,15*(LAB2) + 0,10*(LAB3) + 0,05*LAB4a + 0,05*LAB4b + 0,05*LAB4c + 0,10*LAB4d + 0,20*LAB5 + 0,15*LAB6$$

$$\text{Nota final do semestre (Moodle)} = \text{Media do semestre (Moodle)} = (MP + MT)/2$$

Conforme parágrafo 2º do artigo 70 da Resolução 17/CUn/97, o aluno com frequência suficiente (FS) e média final no período (MF) entre 3,0 e 5,5 terá direito a uma nova avaliação ao final do semestre (REC), sendo a nota final (NF) calculada conforme parágrafo 3º do artigo 71 desta resolução, ou seja: $NF = (MF + REC) / 2$.

Casos de recuperação: se $3,0 \leq \text{Media do semestre} < 6,0$ e frequência 75%,

Nota de recuperação (REC) = Nota da prova de recuperação.

$$\text{Nota final do semestre (Moodle)} = [\text{Media final do semestre (Moodle)} + REC] / 2.$$

Para realização de avaliações em atraso, de acordo com a RESOLUÇÃO Nº 17/CUn/97, de 30 de setembro de 1997:

Art. 70 § 4o - Ao aluno que não comparecer às avaliações ou não apresentar trabalhos no prazo estabelecido será atribuída nota 0 (zero).

Art. 74 - O aluno, que por motivo de força maior e plenamente justificado, deixar de realizar avaliações previstas no plano de ensino, deverá formalizar pedido de avaliação à Chefia do Departamento de Ensino ao qual a disciplina pertence, dentro do prazo de 3 (três) dias úteis, recebendo provisoriamente a menção I.

Cronograma:

11/8 Unidade I - Plano de Ensino, Histórico da Programação concorrente (AT)

13/8 Unidade II - Multithreading: Threads, Estados de threads, Prioridades de threads, Implementação de threads (AT, AP)

18/8 Ambiente de execução Multithreading (AT)

20/8 Sincronização de threads: Produtor-Consumidor sem sincronização (AP)

25/8 Unidade III - Monitores (AT)

27/8 Produtor-Consumidor com sincronização de Monitor (AP)

01/9 Atividade de Laboratório 1 (Lab 1) (TP)

03/9 Atividade de Laboratório 1 (Lab 1) (TP)

08/9 Unidade III - Locks (AT)

10/9 Produtor-Consumidor com sincronização de Looks (AP)

15/9 Atividade de Laboratório 2 (Lab 2) (TP)

17/9 Atividade de Laboratório 2 (Lab 2) (TP)

22/9 Unidade III - Semáforos (AT)

24/9 Atividade de Laboratório 3 (Lab 3) (TP)

29/9 Atividade de Laboratório 3 (Lab 3) (TP)

01/10 Questionário 1 (TT)

06/10 Prova 1 - Assunto: Unidades I, II e III

08/10 Correção da Prova 1

13/10 Unidade IV - Sockets - Aplicações Distribuídas (AT)

15/10 Datagram Sockets (AP)

20/10 TCP Stream Sockets (AP)

22/10 Multicast Socket (AP)

27/10 Atividade Laboratório 4 (Lab 4) (TP)

29/10 Atividade Laboratório 4 (Lab 4) (TP)

03/11 Unidade V - Java RMI - Aplicações Distribuídas (AP)

05/11 11/6 Exemplo básico RMI (AP)

10/11 Atividades do Laboratório 5 (Lab 5) (TP)

12/11 Atividades do Laboratório 5 (Lab 5) (TP)

17/11 Unidade VI - Eventos e Notificações (AT)

19/11 Atividades do Laboratório 6 (AP)

24/11 Questionário 2 (TT)

25/11 Prova 2 - Unidades IV, V e VI

01/12 Correção da Prova 2

03/12 Atividades do Laboratório 6 (TP)

08/12 Atividades do Laboratório 6 (TP)

Bibliografia Básica:

ANDREWS, G. R., Concurrent Programming, Benjamin-Cummings, 1991.

COULOURIS, George; DOLLIMORE, Jean; KINDBERG, Tim. Distributed Systems: -Concepts and Design. 3rd Edition. Addison-Wesley, 2001.

DEA, Doug “Concurrent Programing in Java”, 2nd Ed., Addison-Wesley, 2000.

DEITEL, Harvey M.; DEITEL, Paul J. Java: Como Programar. 4a Edição. Bookman, 2002.

HORSTMANN, Cay S.; CORNELL, Gary. Core Java 2. Vol I e II. Makron Books, 1999.

Bibliografia Complementar:

ORFALI, Robert; HARVEY, Dan. Client/Server Programming with Java and CORBA. 2nd Edition. John Wiley, 1998.

STEVENS, W. R., Unix Network Programming, vols. 1 e 2, Prentice-Hall, 1998.

TANENBAUM, Andrew. Sistemas Operacionais Modernos. Prentice-Hall, 2003.