



## Plano de Ensino

---

### 1) Identificação

<b>Disciplina:</b>	INE5645 - Programação Paralela e Distribuída		
<b>Turma(s):</b>	0538A, 0538B		
<b>Carga horária:</b>	72 horas-aula	Teóricas: 36	Práticas: 36
<b>Período:</b>	2º semestre de 2009		

### 2) Cursos

- Sistemas de Informação (238)

### 3) Requisitos

- INE5611 - Sistemas Operacionais

### 4) 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.

### 5) Objetivos

Geral: 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

### 6) Conteúdo Programático

- 6.1) Introdução à Programação Paralela e Distribuída [4 horas-aula]
  - Vantagens e Dificuldades
  - Plataformas de Execução
  - Suporte Computacional
- 6.2) Programação Paralela [16 horas-aula]
  - Processos
  - Threads
  - Paralelismo em Java
- 6.3) Controle de Concorrência [18 horas-aula]
  - Monitores
  - Locks
  - Semáforos
  - Concorrência na API Java
- 6.4) Programação Distribuída [8 horas-aula]
  - Modelos de sistemas distribuídos
  - Elementos básicos da comunicação
  - Comunicação por passagem de mensagem
- 6.5) Comunicação entre Processos [26 horas-aula]
  - Pipes
  - Sockets
  - RMI

### 7) Metodologia

Serão ministradas aulas expositivas sobre os conceitos relacionados à programação concorrente, paralela e distribuída, utilizando notebook e projetor multimídia.

Serão realizadas atividades práticas em laboratório para o aprendizado destes conceitos utilizando a linguagem Java.

Um projeto final deverá ser desenvolvido pelos alunos, utilizando uma ou mais tecnologias estudadas ao longo

da disciplina.

## 8) Avaliação

Os alunos serão avaliados com base em seu desempenho nas seguintes atividades: 2 Provas teóricas (P1 e P2); 4 Atividades de laboratório (LAB1 a 4); 1 Projeto final (PROJ).

A média teórica será calculada aplicando a fórmula:  $MT = (P1 + P2)/2$ .

Caso o aluno não alcance média teórica 6,0 (seis), uma prova de recuperação (PR) será aplicada. A média MT, para esse caso, será  $((P1+P2)/2+PR)/2$ .

A média prática será atribuída aplicando a fórmula  $MP = (LAB1 + LAB2 + LAB3 + LAB4 + 2 * PROJ)/6$ .

Para ser aprovado, o aluno deve possuir médias teórica e prática iguais ou superiores a 6,0 (seis). Nesse caso, a média final será calculada aplicando a fórmula  $MF = (MT + MP)/2$ . Caso contrário, a média final será igual ao valor mais baixo dentre as médias teórica e prática.

Observações:

Será atribuída uma nota para a equipe para cada trabalho de laboratório. Se um aluno não responder a uma pergunta do trabalho, a equipe terá desconto na nota do trabalho. Em caso de cópia de trabalhos de qualquer fonte, todos os alunos envolvidos terão nota igual a zero. Será utilizada uma ferramenta de detecção de plágio (ex. SIM - Software and text similarity tester) para comparação dos trabalhos.

Trabalhos entregues com atraso terão desconto automático de pontos. Após a segunda semana (14 dias após o fim do prazo original), o trabalho não mais será aceito, ou seja, terá nota zero.

Em caso de "cola" nas avaliações, os alunos envolvidos estarão automaticamente reprovados com média zero, e o caso será reportado ao colegiado do curso para que sejam tomadas medidas disciplinares.

**OBS:** dado que a disciplina apresenta pelo menos 50% de sua carga ocupada por aulas práticas ou pelo menos 50% da nota final obtida com avaliações práticas, conforme deliberação do Colegiado do Curso de Sistemas de Informação de 31 de julho de 2008, ela não prevê a realização de avaliação no final do semestre (recuperação) de que trata o parágrafo 2º do artigo 70 da Resolução 17/CUn/97.

## 9) Cronograma

1a Semana: Introdução à Programação Paralela e Distribuída

2a Semana: Programação Paralela

3a Semana: Controle de Concorrência - Monitores

3a e 4a Semana: Atividade de laboratório I

5a Semana: Controle de Concorrência - Lock

5a e 7a Semana: Atividade de laboratório II

8a Semana: Controle de Concorrência - Semáforos

8a e 10a Semana: Atividade de laboratório III

10a Semana: Prova I

11a Semana: Programação Distribuída

12a Semana: Comunicação entre Processos - Socket UDP, TCP e IP Multicast

12a e 14a Semana: Atividade de laboratório IV

15a Semana: Comunicação entre Processos - Java RMI

16a Semana: Prova II

17a e 18a Semana: Projeto Final

## 10) 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 &#8220;Concurrent Programing in Java&#8221;, 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.

## 11) 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.