

Escola/Campus:	Escola Politécnica/Curitiba		
Curso:	Bacharelado em Ciência da Computação (BCC)	Ano/Semestre:	2024/1
Código/Nome da disciplina:	Arquitetura de Sistemas Distribuídos, Paralelos e Concorrentes (ASDPC)		
Carga Horária:	120 horas-aula		
Requisitos:	Programação Distribuída, Paralela e Concorrente		
Créditos:		Período: 7	Turma: U
Professor Responsável:	Luiz Augusto de Paula Lima Jr.		
	Turno: Manhã e Noite		

1. Ementa:

Esta disciplina, de natureza teórico-prática, é ofertada no sétimo período do Bacharelado em Ciência da Computação e destina-se a estudantes que desejam projetar e desenvolver aplicações distribuídas, paralelas e concorrentes usando algoritmos e plataformas distribuídas avançadas satisfazendo às requisições do sistema final visado, inclusive na presença de falhas. Mais especificamente, ao final da disciplina, o estudante será capaz de projetar e implementar algoritmos distribuídos, paralelos e concorrentes sobre diferentes classes de plataformas desenvolvendo, desta maneira, aplicações eficientes e robustas, mesmo em cenários reais suscetíveis a falhas. Para o melhor aproveitamento da disciplina é necessário conhecimento prévio de aspectos básicos de programação distribuída, paralela e concorrente.

2. Relação com disciplinas precedentes e posteriores

Esta disciplina é parte integrante do eixo de programação do curso, fornecendo a base conceitual e tecnológica necessária para a programação de sistemas computacionais complexos e de grande porte. Ela possui como pré-requisito a disciplina de Programação Distribuída, Paralela e Concorrente (3º período) empregando os resultados de aprendizagem desta última e contribuindo com as disciplinas de Complexidade de Algoritmos (7º. Período), porém considerando algoritmos distribuídos, paralelos e concorrentes e de Avaliação de Desempenho de Sistemas (8º. Período), na validação experimental de resultados obtidos analiticamente.

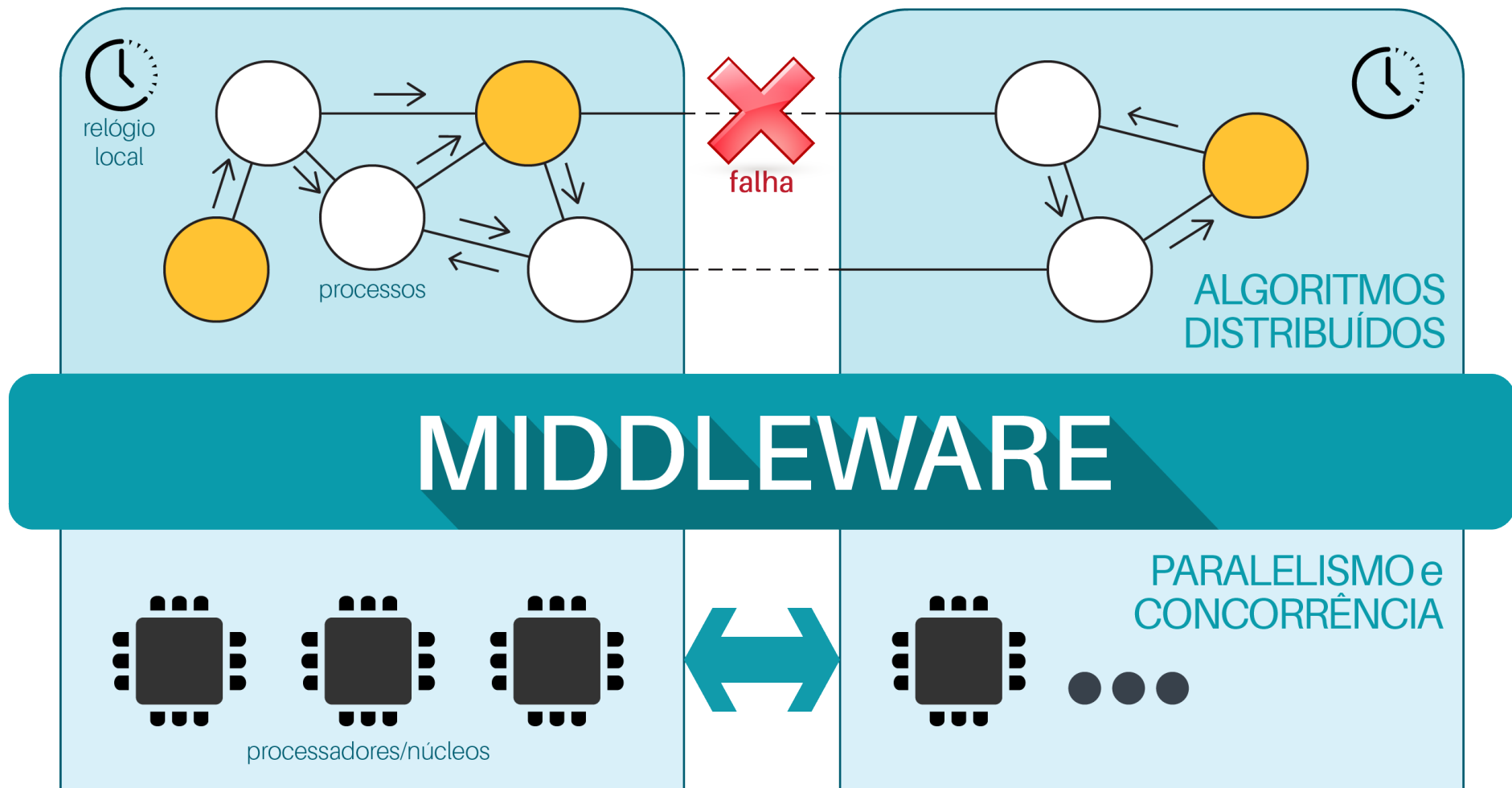
3. Temas de estudo

- **TE01** - Conceitos Fundamentais (heterogeneidade e interoperabilidade, tendências, paradigmas e tunelamento)
- **TE02** - Coordenação Distribuída de Processos
- **TE03** - *Middleware* (princípios, classes, estudos de casos)
- **TE04** - Algoritmos Distribuídos (projeto, análise e implementação)
- **TE05** - Computação na Presença de Falhas

4. Resultados de Aprendizagem

Competência	Elemento de Competência	Resultado de Aprendizagem	Temas de Estudo
Projetar infraestrutura computacional sustentável, com segurança e dependabilidade, considerando tecnologias, estrutura organizacional e plano diretor de tecnologia da informação, implantando e monitorando sua execução de forma ética e resiliente.	Planejar infraestrutura computacional sustentável e compartilhada para sistemas computacionais a partir de suas especificações	RA1. Projetar algoritmos distribuídos, paralelos e concorrentes eficientes de acordo com os requisitos da aplicação e com as condições do ambiente computacional	TE04
	Implantar a infraestrutura computacional, com domínio do processo de aquisição ou contratação de componentes de hardware e software, bem como do processo de instalação, configuração e integração desses componentes, levando em conta os preceitos éticos do relacionamento comercial	RA2. Desenvolver sistemas computacionais distribuídos, paralelos e concorrentes usando plataforma adequada de acordo com os requisitos da aplicação e com as condições do ambiente computacional	TE03
	Manter a infraestrutura computacional em conformidade com a sua especificação	RA3. Aplicar técnicas de coordenação distribuída de processos e tolerância a faltas para a concepção de aplicações corretas, eficientes e robustas	TE01 TE02 TE05

5. Mapa Mental



6. Metodologia e Avaliação

Alinhamento Construtivo			
Resultado de aprendizagem	Indicadores de desempenho	Processos de Avaliação	Métodos ou técnicas empregados**
RA1. Projetar algoritmos distribuídos, paralelos e concorrentes eficientes de acordo com os requisitos da aplicação e com as condições do ambiente computacional	ID1.1: Identifica algoritmo distribuído adequado para a solução de um problema computacional específico ID1.2: Analisa a complexidade/eficiência de algoritmo distribuído ID1.3: Implementa algoritmo distribuído usando plataformas adequadas ID1.4: Compõe algoritmos distribuídos fundamentais para a solução de problemas complexos	PBL individual, com avaliação formativa e <i>feedback</i> coletivo Avaliação somativa individual e <i>feedback</i> PjBL em equipe, com avaliação somativa e <i>feedback</i> no momento da defesa	Aula expositiva dialogada PBL/PjBL Exercícios <i>Será utilizado o ambiente virtual de aprendizagem para organização do material didático e para a interação com os estudantes.</i>
RA2. Desenvolver sistemas computacionais distribuídos, paralelos e concorrentes usando plataforma adequada de acordo com os requisitos da aplicação e com as condições do ambiente computacional	ID2.1: Implementa sistema distribuído possivelmente heterogêneo usando diferentes classes de middlewares ID2.2: Utiliza recursos de plataformas distribuídas, paralelas e concorrentes na construção de sistemas distribuídos ID2.3: Integra e configura componentes de um sistema distribuído de acordo com os requisitos e as condições de um sistema computacional	PBL individual, com avaliação formativa e <i>feedback</i> coletivo Avaliação somativa individual e <i>feedback</i> PjBL em equipe, com avaliação somativa e <i>feedback</i> no momento da defesa	Aula expositiva dialogada PBL/PjBL Exercícios <i>Será utilizado o ambiente virtual de aprendizagem para organização do material didático e para a interação com os estudantes.</i>
RA3. Aplicar técnicas de coordenação distribuída de processos e tolerância a faltas para a concepção de aplicações corretas, eficientes e robustas	ID3.1: Utiliza corretamente ferramentas de coordenação distribuída de processos em um sistema distribuído ID3.2: Aplica técnicas de tolerância a falta de forma a melhorar robustez do sistema	PBL individual, com avaliação formativa e <i>feedback</i> coletivo Avaliação somativa individual e <i>feedback</i> PjBL em equipe, com avaliação somativa e <i>feedback</i> no momento da defesa	Aula expositiva dialogada PBL/PjBL Exercícios <i>Será utilizado o ambiente virtual de aprendizagem para organização do material didático e para a interação com os estudantes.</i>

PBL: *Problem-Based Learning* ⇒ trabalho prático de curta duração, feito individualmente

PjBL: *Project-Based Learning* ⇒ trabalho prático de longa duração, feito em equipe

Para a **aprovação** na disciplina é necessário obter uma **média ponderada – M**, conforme tabela abaixo – maior ou igual a 7,0 pontos. Será oportunizada uma **recuperação parcial** para estudantes que não atingirem média 7,0 ao longo semestre. **Só terão direito a esta recuperação parcial estudantes que tiverem entregue todos trabalhos previstos na disciplina** (excluindo-se TDE). Após o término do semestre, ainda haverá a semana estendida de **recuperação**. Têm direito a esta **recuperação final** somente estudantes com média **M**, tal que $4,0 \leq M < 7,0$. Todas as recuperações serão **por RA** em que o estudante não atingiu nota mínima 7,0, e a nota obtida será truncada em 7,0 pontos. Caso seja necessária recuperação final, a nota final máxima do estudante será limitada a 7,0 pontos após a aplicação da ponderação por RA.

Abaixo, a tabela com os **trabalhos** (T_i), **TDE** e avaliações **somativas** (S_i) e a sua contribuição para cada RA, e a ponderação de cada RA na média final **M**.

	RA1	RA2	RA3
M	40%	40%	20%
T1	30%	20%	
T2		10%	20%
TDE	10%	10%	20%
S1		30%	30%
S2	60%	30%	30%

7. Cronograma de atividades

S	Data	Atividade Pedagógica	CH
1	19,21/2	Motivação e introdução à disciplina de ASDPC. Objetivos e tendências. Conceitos Fundamentais: heterogeneidade e interoperabilidade e aspectos de projeto. Introdução a Tolerância a Faltas.	6
2	26,28/2	Arquiteturas e Paradigmas de Comunicação (filtros, C/S, P2P, DSM). Fundamentos de comunicação. Pipes, ferramentas de comunicação básica (sockets). Experimentos e exercício prático e feedback.	6
3	4,6/3	Coordenação Distribuída de Processos Coordenação Distribuída de Processos na prática. Início TDE	6+2
4	11,13/3	<i>Middleware</i> s — princípios fundamentais e classificação Objetos Distribuídos: conceitos, arquitetura e experimentos	6+2
5	18,20/3	Objetos Distribuídos: desenvolvimento C/S básico. Objetos Distribuídos: aspectos avançados (NS, políticas e padrões). Início T1	6+2
6	25,27/3	OD: Exercícios adicionais e feedback	6+2
7	1,3/4	Web Services: conceitos e arquitetura Web Services-SOAP: experimentos e exercícios e feedback.	6+2
8	8,10/4	Web Services-REST Web Services-REST: experimentos e exercícios e feedback.	6+2
9	15,17/4	Defesas de Projeto T1 - Fim T1 Teste teórico somativo 1	6+2
10	22,24/4	MOM, DSM e Grid. O protocolo AMQP. AMQP: experimentos práticos e exercícios. Algoritmos Distribuídos: motivação — PjBL: broadcast - Fim TDE	6
11	29/4	Ambientes Computacionais Distribuídos: introdução, formalização e conceitos fundamentais. Feedback implementação do broadcast. Flooding e Wakeup Implementação prática. Exercício e <i>feedback</i> .	4
12	6,8/5	Percurso Distribuído. Computação distribuída em árvores: motivação e construção de <i>spanning trees</i> . Exercício e feedback. Início T2	6
13	13,15/5	Feedback implementação do algoritmo distribuído de construção de <i>spanning trees</i> . Tempo em Sistemas Distribuídos	6
14	20,22/5	Computação Distribuída na Presença de Falhas: Motivação via PBL. TMR. Replicação e Consistência de Dados e Consenso Distribuído	6
15	27,29/5	Defesas de Projeto T2 - Fim T2 Teste teórico somativo 2	6
16	3,5/6	Feedback e esclarecimento de dúvidas	6
17	10,12/6	Atividade de Recuração de RAs	6
18	17,19/6	Revisão, esclarecimento de dúvidas e preparação para recuperação final	6
	24/6	Recuperação da Semana Estendida	

8. Bibliografia

Todas as atividades contarão com o apoio de material preparado pelo professor sob a forma de slides, vídeos e exemplos de código disponibilizados no ambiente virtual de aprendizagem. Os seguintes livros poderão ser consultados na biblioteca da PUCPR pelos estudantes como fonte complementar:

8.1. Básica

1. TANENBAUM, A. S.; STEEN, M. **Sistemas Distribuídos: Princípios e Paradigmas**. 2a Edição, Prentice Hall, 2007.
2. COULOURIS, George; DOLLIMORE, Jean; KINDBERG, Tim. **Sistemas Distribuídos: Conceitos e Projeto**. Ed. Bookman, 2007.
3. ANDREWS, Gregory R. **Foundations of multithreaded, parallel, and distributed programming**. Massachusetts: Addison-Wesley, 2000. xx, 664 p. ISBN 0-201-35752-6.
4. LYNCH, Nancy A. **Distributed Algorithms**. San Francisco: M. Kaufmann 1997.

8.2. Complementar

1. SANTORO, N. **Design and Analysis of Distributed Algorithms**, John Wiley & Sons, 2006.
2. FOKKINK, W. **Distributed Algorithms: An Intuitive Approach**. The MIT Press, 2013.
3. BEN-ARI, M. **Principles of concurrent and distributed programming**. 2nd ed. Harlow, England: Addison-Wesley, 2006. xv, 361 p ISBN 0-321-31283-X (broch.).
4. Butenhof, D. R. (2006) **Programming with POSIX Threads**, Addison-Wesley, 2006.
5. Downey, A. B. **The Little Book of Semaphores**, Second Edition, 2016.

9. Acessibilidade

Não há necessidade de adaptação.

10. Adaptações para práticas profissionais

As atividades práticas previstas nesta disciplina podem ser feitas com acesso remoto a um servidor da PUCPR onde está instalado todo o software necessário para realizarem trabalhos, projetos e desenvolvimentos previstos.