

# Laboratório de Programação II

**Luciano Moraes Da Luz Brum**

**Universidade Federal do Pampa – Unipampa – Campus Bagé**

**Email:** [lucianobrum18@gmail.com](mailto:lucianobrum18@gmail.com)

**Site:** <https://sites.google.com/view/brumluciano/página-inicial>

# Apresentação

- **Horários:** Sexta (períodos 3 e 4 Noturno);
- **Carga horária:** 40 horas, 4 créditos (2 Teóricos + 2 Semipresenciais);
- **Professor:** Luciano Moraes Da Luz Brum
- **Email:** [lucianobrum18@gmail.com](mailto:lucianobrum18@gmail.com)
- **Site:** <https://sites.google.com/view/brumluciano/página-inicial>
- **Entrega de trabalhos:** Site da disciplina ou por email.

# Ementa

- Funções especiais de uma linguagem de programação.
- Grafos: algoritmos de caminhamento, árvore geradora, caminho mínimo e planaridade.
- Implementação de algoritmos avançados de grafos.
- Criptografia e outros problemas clássicos da computação.



# Objetivo

- Fornecer conhecimentos práticos e avançados sobre uma linguagem de programação através da implementação de estruturas e algoritmos clássicos da computação.

# Metodologia

- A disciplina é prática e será executada com base em trabalhos práticos de programação na forma de projetos, em grupos e/ou individuais, para a programação de problemas computacionais na área de ponteiros e estruturas genéricas, grafos, criptografia e outros problemas. As aulas ocorrerão no laboratório de informática.
- Técnicas e Recursos necessários:
- Aulas expositivas: Quadro e/ou software de apresentação e datashow.
- Proposta de atividades de programação: Computador com compilador e/ou ambiente de programação adequado instalado.

# Avaliação

- A média semestral (MS) do aluno será composta por:
  - 5 (cinco) atividades de programação (AP1 - AP5). Cada atividade terá o peso de 20% na média semestral.



# Avaliação

- Para o cálculo da média semestral (MS), basta calcular a média aritmética das notas de cada avaliação (5 avaliações). A expressão para o cálculo da média semestral é dado como:  $MS = (AP1+AP2+AP3+AP4+AP5)/5$
- Os conteúdos das avaliações são acumulativos, ou seja, a cada avaliação será solicitado conhecimentos de disciplinas anteriores e da própria disciplina.

# Avaliação

- Para serem aprovados, os alunos deverão obter média semestral maior ou igual a 6,0 (seis), sendo, neste caso, a média final do aluno.
- Se necessário, o aluno poderá realizar a prova de recuperação final.
- Para serem aprovados os acadêmicos devem obter uma média maior ou igual a 6,0 (seis) entre a nota da prova de recuperação final e a média semestral obtida até então, sendo que esta será a média final para estes acadêmicos.



# Avaliação

- Para a aprovação é imprescindível 75% de presença às aulas da disciplina por parte dos alunos. Alunos com menos de 75% de presença estão automaticamente reprovados, independente das notas obtidas.
- As atividades semipresenciais, que neste caso serão as próprias atividades de programação, também fazem parte da frequência do aluno e serão contabilizadas de acordo com o número de horas descritas para cada atividade.

# Avaliação

- Avaliação de Recuperação:
  - Provas curtas (poucas questões).
  - Realização de trabalhos de pesquisa e/ou programação e/ou modelagem.
  - Provas orais.
  - Listas de exercícios.
  - Revisão das atividades realizadas nas próprias avaliações (refazer partes com problemas, no caso de atividades de programação).
  - Estas atividades de recuperação serão agendadas com os alunos no momento oportuno.

**Cronograma e Programa do Componente Curricular - Presencial**

<b>Data</b>	<b>Número da aula</b>	<b>Carga horária</b>	<b>Tipo</b>	<b>Conteúdos/Descrição</b>
10/03/2017	1	55 Min	Prática	Apresentação de ementa e plano de ensino.
10/03/2017	2	55 Min	Prática	Revisão de conteúdo de disciplinas anteriores através de uma proposta de atividade.
17/03/2017	3	55 Min	Prática	Ponteiros genéricos e estruturas genéricas. Proposta de atividade envolvendo o uso de ponteiro void * (lista encadeada). Atividade de programação I.
17/03/2017	4	55 Min	Prática	Ponteiros genéricos e estruturas genéricas. Proposta de atividade envolvendo o uso de ponteiro void * (lista encadeada). Atividade de programação I.
24/03/2017	5	55 Min	Prática	Ponteiros genéricos e estruturas genéricas.
24/03/2017	6	55 Min	Prática	Ponteiros genéricos e estruturas genéricas.
31/03/2017	7	55 Min	Prática	Implementação de uma árvore AVL genérica: Atividade de programação II.
31/03/2017	8	55 Min	Prática	Implementação de uma árvore AVL genérica: Atividade de programação II.
07/04/2017	9	55 Min	Prática	Implementação de uma árvore AVL genérica.
07/04/2017	10	55 Min	Prática	Implementação de uma árvore AVL genérica.
28/04/2017	11	55 Min	Prática	Introdução a Grafos: Formas de representação, características e aplicações. Algoritmos elementares em grafos: DFS e BFS.
28/04/2017	12	55 Min	Prática	Introdução a Grafos: Formas de representação, características e aplicações. Algoritmos elementares em grafos: DFS e BFS.
05/05/2017	13	55 Min	Prática	Grafos: ordenação topológica e árvore de amplitude mínima (Kruskal e Prim). Atividade de programação III.
05/05/2017	14	55 Min	Prática	Grafos: ordenação topológica e árvore de amplitude mínima (Kruskal e Prim). Atividade de programação III.
12/05/2017	15	55 Min	Prática	Grafos: ordenação topológica e árvore de amplitude mínima (Kruskal e Prim).
12/05/2017	16	55 Min	Prática	Grafos: ordenação topológica e árvore de amplitude mínima (Kruskal e Prim).



19/05/2017	17	55 Min	Prática	Grafos: ordenação topológica e árvore de amplitude mínima (Kruskal e Prim).
19/05/2017	18	55 Min	Prática	Grafos: ordenação topológica e árvore de amplitude mínima (Kruskal e Prim).
26/05/2017	19	55 Min	Prática	Grafos: algoritmos de menor caminho (Dijkstra) - Atividade de programação IV.
26/05/2017	20	55 Min	Prática	Grafos: algoritmos de menor caminho (Dijkstra) - Atividade de programação IV.
02/06/2017	21	55 Min	Prática	Grafos: algoritmos de menor caminho (Dijkstra).
02/06/2017	22	55 Min	Prática	Grafos: algoritmos de menor caminho (Dijkstra).
09/06/2017	23	55 Min	Prática	Grafos: algoritmos de menor caminho (Dijkstra).
09/06/2017	24	55 Min	Prática	Grafos: algoritmos de menor caminho (Dijkstra).
23/06/2017	25	55 Min	Prática	Criptografia: conceitos básicos, métodos. Atividade de programação V.
23/06/2017	26	55 Min	Prática	Criptografia: conceitos básicos, métodos. Atividade de programação V.
30/06/2017	27	55 Min	Prática	Criptografia: conceitos básicos, métodos.
30/06/2017	28	55 Min	Prática	Criptografia: conceitos básicos, métodos.
07/07/2017	29	55 Min	Prática	Data limite para entrega da atividade de programação V e qualquer outra atividade (ou parte de) que por ventura não tenha sido entregue ou que tenha necessidade de revisão.

# Atividades Semipresenciais

07/07/2017	30	55 Min	Prática	Data limite para entrega da atividade de programação V e qualquer outra atividade (ou parte de) que por ventura não tenha sido entregue ou que tenha necessidade de revisão.
14/07/2017	31	55 Min	Prática	Prova de Recuperação Final.
14/07/2017	32	55 Min	Prática	Prova de Recuperação Final.
17/02/2017	33	55 Min	Prática	Correção da Recuperação e entrega das notas finais.

## Cronograma e Programa do Componente Curricular - Semipresencial

Data	Carga Horária	Conteúdos/Descrição
31/03/2017	4.00 Horas	Estruturas Genéricas e Ponteiros Genéricos.
21/04/2017	4.00 Horas	Árvore AVL Genérica.
26/05/2017	8.00 Horas	Grafos: ordenação topológica e árvore de amplitude mínima (Kruskal e Prim).
23/06/2017	7.00 Horas	Grafos: algoritmos de menor caminho (Dijkstra).
07/07/2017	7.00 Horas	Atividades envolvendo criptografia.

# Atendimentos e Bibliografia

## Atendimento aos Acadêmicos

Dia da Semana	Início	Final	Local	Observação
Sexta-feira	18:00:00	20:40:00	Sala 3143	Em horários alternativos, agendar com o professor por e-mail (lucianobrum18@gmail.com) ou procurar o professor em sua sala.

## Ações Interdisciplinares entre Ensino-Pesquisa-Extensão

Não estão previstos projetos interdisciplinares para este componente curricular.

## Outras Ações

Não estão previstas visitas técnicas específicas para este componente curricular.

## Bibliografia Básica

CORMEN, Thomas H.; et al. Algoritmos: teoria e prática. Rio de Janeiro: Campus, 2002. 916 p.

SCHILD, Herbert. C. Completo e Total. 3 ed. São Paulo: Makron Book, 1997.

ZIVIANI, Nívio. Projetos de Algoritmos: com implementação em Pascal e C. 2ª Ed. Revista e Ampliada. São Paulo: Pioneira Thomson Learning , 2004.

## Bibliografia Complementar

DROZDEK, Adam. Estrutura de Dados e Algoritmos em C++. São Paulo: Pioneira Thomson Learning, 2002. 579 p.

CELES, Waldemar; CERQUEIRA, Renato; RANGEL, José Lucas. Introdução a Estruturas de Dados com técnicas de programação em C. Rio de Janeiro: Elsevier (Campus), 2004. 4ª Reimpressão. 294 p.

VELOSO, Paulo; SANTOS, Clésio dos; AZEREDO, Paulo; FURTADO, Antonio. Estrutura de Dados. Rio de Janeiro: Campus, 1996.

TENENBAUM, Aarón M.; LANGSAM, Y.; AUGENSTEIN, M. J. Estruturas de Dados usando C. São Paulo: Makron Books, 1995. 884 p.

KERNIGHAN, Brian W.; RITCHIE, Dennis M. C: A Linguagem de Programação Padrão Ansi. Editora Campus.





DÚVIDAS ?