



BY

“C faz com que dar um tiro no pé seja fácil; C++ torna isso mais difícil, mas quando nós o fazemos arrebetamos com a perna toda” (Bjarne Stroustrup).

# Orientação a Objetos

Paulo Ricardo Lisboa de Almeida



# Professor



Paulo Ricardo Lisboa de Almeida  
Doutor em Ciência da Computação, Bel. em  
Eng. da Computação.

[paulorla@ufpr.br](mailto:paulorla@ufpr.br)

[prl Almeida.com.br](http://prl Almeida.com.br)

[www.linkedin.com/in/paulorla](http://www.linkedin.com/in/paulorla)

# Professor - Pesquisa

Grupo DSBD.

Aprendizado de máquina.

Tratamos mais sobre o assunto no final da disciplina.

Iniciação científica, TCC, Mestrado, Doutorado, ...

Cidades inteligentes.



# Professor - Pesquisa

Concept Drifts.

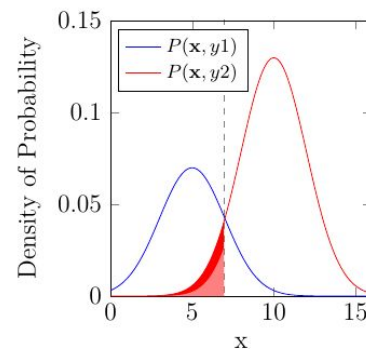
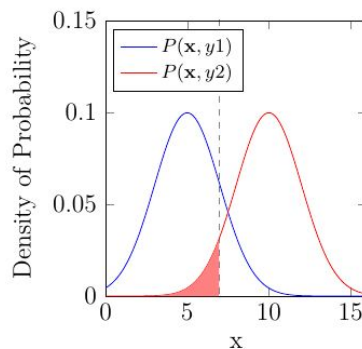
Os ambientes do mundo real são dinâmicos.

Estão sempre mudando.

Como criar um IA capaz de identificar que o ambiente mudou?

Se adaptar automaticamente.

Como criar benchmarks para esses cenários?



# A Disciplina

Orientação a Objetos

Códigos de Disciplina

Graduação: Prática em Desenvolvimento de Software - CI1353, CI214

Pós-Graduação: Tópicos Especiais I - INFO7070

Objetivo principal

Introduzir conceitos básicos relacionados a orientação a objetos de maneira prática

# Ementa

- Histórico.
- Classes e objetos.
- Funções e dados membros.
- Modificadores de acesso.
- Construtores e destrutores.
- Alocação dinâmica de memória.
- Herança e polimorfismo.
- Templates. Bibliotecas padrão.
- Lançamento e tratamento de exceções.

# Moodle

A disciplina tem uma página oficial no Moodle.

Será o meio oficial de comunicação.

Atualize o seu e-mail no Moodle.

Para se comunicar.

Utilize preferencialmente o fórum do Moodle.

Você também pode falar através do chat do Moodle, ou enviar e-mail.

# Exercícios semanais

Exercícios semanais entregues via Moodle.

Não serão aceitas entregas fora do prazo.

Considerando uma nota entre 0 e 100.

Quem fizer pelo menos 90% dos exercícios, ganha 5 pontos extras.

Quem fizer pelo menos 75% dos exercícios, ganha 3 pontos extras.

Quem fizer pelo menos 50% dos exercícios, ganha 2 pontos extras.

Indício de plágio e afins em qualquer exercício, trabalhos e provas fazem com que o aluno **perca automaticamente o direito a qualquer nota extra.**



# Avaliação

40%: Trabalho final.

30%: Prova 1

30%: Prova 2

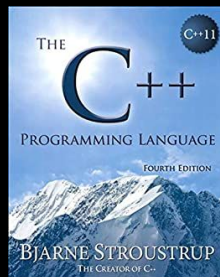
**Mínimo** de presenças: 75% -> **reprovado** automaticamente caso contrário.

**Aprovado** se média maior  $\geq$  que 70.

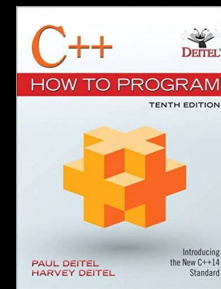
**Exame** se nota  $\geq$  40. Nesse caso nota = (média + exame)/2. **Aprovado** se nota  $\geq$  5.

# Bibliografia Básica

Bjarne Stroustrup. The C++ Programming Language. Addison-Wesley, 2013.



Deitel, H. M., Deitel, P. J. C++: como programar. 5a ed. Pearson Prentice Hall. 2006.

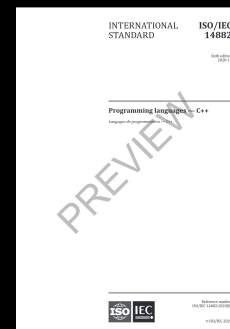


Gamma, E. Padrões de Projetos: Soluções Reutilizáveis. Bookman. 2009.



ISO/IEC 14882:2020 Programming languages - C++:  
[www.iso.org/obp/ui/#iso:std:iso-iec:14882:ed-6:v1:en](http://www.iso.org/obp/ui/#iso:std:iso-iec:14882:ed-6:v1:en)

Draft público em  
<https://isocpp.org/files/papers/N4860.pdf>



# Ficha 2

Para mais detalhes, como bibliografia complementar e programa completo da disciplina, veja a Ficha 2.

É um plano, portanto pode sofrer alterações.

Em especial, as datas podem mudar.

Tudo será avisado com antecedência.

# Entregas e Provas

Não serão aceitas entregas em atraso (exceto casos amparados pela UFPR).

Em caso de plágio, **todos** envolvidos ficam com zero.

# Bibliografia - Internet

**Cuidado!**

A internet é uma fonte importante de informações.

# Bibliografia - Internet

## **Cuidado!**

A internet é uma fonte importante de informações.

E uma fonte inesgotável de bobagens e pseudoespecialistas!

Seja criterioso ao pesquisar algum conceito na internet.

Na dúvida entre em contato com o professor.

# Conhecimentos e Equipamentos

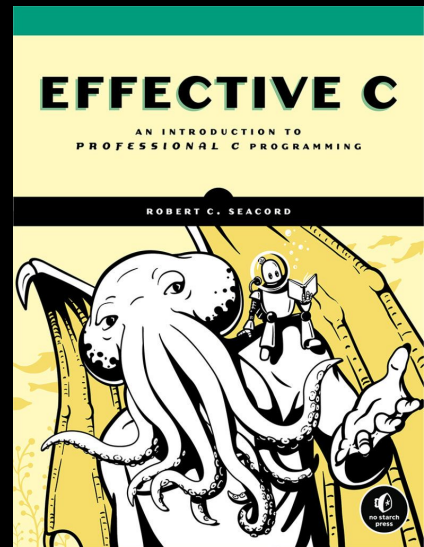
Você precisa ter o domínio da **linguagem C**.

Structs, ponteiros, alocação dinâmica, ...

Se você precisa rever esses conceitos, um bom livro é:

Seacord, R. C. *Effective C: An Introduction to Professional C Programming*. 2020.

Você vai precisar também de um computador, e do g++.



# Importância

Softwares orientados a objetos são comuns em todas as áreas, principalmente para softwares grandes e complexos.

Exemplos em Machine Learning.

OpenCV -> C++

Weka -> Java

MOA -> Java



# Importância

Na disciplina vamos utilizar a linguagem **C++** como **ferramenta** para aplicação prática das ideias.

Sempre que possível, faremos paralelos com outras linguagens O.O., como Java e C#.



Criado por Bjarne Stroustrup in 1979.  
Originalmente “C with classes”.



Criado por Bjarne Stroustrup in 1979.  
Originalmente “C with classes”.

Extensão do C.

C++ é compatível com C.



Criado por Bjarne Stroustrup in 1979.  
Originalmente “C with classes”.

Extensão do C.  
C++ é compatível com C.

Padrão ISO.

C++

Criado por Bjarne Stroustrup in 1979.  
Originalmente “C with classes”.

Extensão do C.  
C++ é compatível com C.

Padrão ISO.

Compilado.



Criado por Bjarne Stroustrup in 1979.  
Originalmente “C with classes”.

Extensão do C.  
C++ é compatível com C.

Padrão ISO.

Compilado.

Portável (independente de hardware via compiladores).



Criado por Bjarne Stroustrup in 1979.  
Originalmente “C with classes”.

Extensão do C.  
C++ é compatível com C.

Padrão ISO.

Compilado.

Portável (independente de hardware via compiladores).

Fortemente tipado.



Assim como o C, o C++ dá grande controle do hardware e flexibilidade.

**Não espere** que, como no Java, o programa avise quando você tentar acessar um elemento inexistente em um vetor.

C++ adiciona os **conceitos de O.O.** em C com um **baixo overhead**.





Assim como o C, o C++ dá grande controle do hardware e flexibilidade.

**Não espere** que, como no Java, o programa avise quando você tentar acessar um elemento inexistente em um vetor.

C++ adiciona os **conceitos de O.O.** em C com um **baixo overhead**.

**C++ é uma linguagem que espera que o programador saiba o que está fazendo.**



“C faz com que dar um tiro no pé seja fácil; C++ torna isso mais difícil, mas quando nós o fazemos arrebetamos com a perna toda” (Bjarne Stroustrup).



Não espere usar C++ para desenvolver “sistemas de padaria”.

Você pode fazê-lo, mas vai dar muito trabalho.

Para sistemas simples, utilize linguagens mais produtivas (Python, PHP, Java, ...).

# C++17

Durante a disciplina vamos usar a especificação ISO C++17, ou mais recente.

# Alguns softwares atuais feitos em C++

- **Navegadores**
  - Mozilla Firefox
  - Google Chrome
- **Escritório**
  - Microsoft Office
  - Apache OpenOffice
  - Suíte Adobe
- **Sistemas de Banco de Dados**
  - DuckDB
  - Oracle
  - MySQL

# Alguns softwares atuais feitos em C++

- **Motores de Jogos**

- Unreal Engine 1, 2, 3, 4 e 5.
  - Série Batman, Série Borderlands, Bioshock Infinite.
- Source Engine 1 e 2.
  - Half-Life 2, Counter-Strike GO, Half-Life: Alyx, Dota 2.
- CryEngine.
  - Far Cry, Série Crysis.
- id Tech 7
  - Série Doom, Dishonored, Wolfenstein.
- ...

# Alguns softwares atuais feitos em C++

- **Motores de Jogos**

- Unreal Engine 1, 2, 3, 4 e 5.
  - Série Batman, Série Borderlands, Bioshock Infinite.
- Source Engine 1 e 2.
  - Half-Life 2, Counter-Strike GO, Half-Life: Alyx, Dota 2.
- CryEngine.
  - Far Cry, Série Crysis.
- id Tech 7
  - Série Doom, Dishonored, Wolfenstein.

- O desenvolvimento de jogos com gráficos realistas geralmente envolve a construção de motores gráficos utilizando C++.



# Atenção

Será imprescindível que todos cumpram suas tarefas antes de cada aula.

Exercícios, projetos base, leitura de texto.

Durante as aulas, será assumido que tudo foi feito.

**O conteúdo será construído** a partir do que foi solicitado na aula anterior.



# Exercícios

1. Instale o g++ e as bibliotecas necessárias na sua máquina.
2. Compile o programa “olá mundo” de exemplo para testar se sua instalação está correta
3. Crie um programa em C, que deve ter o seguinte:
  - a. Uma struct para representar uma pessoa, com os seguintes dados: nome, cpf e idade.
  - b. Uma função chamada validarCPF, que recebe um número de cpf, e devolve verdadeiro ou falso, indicando se o cpf é válido ou não.
    - i. Pesquise na internet sobre como validar um cpf (de um jeito eficiente – gravar um cpf em um vetor é absurdo!).
  - c. No main crie um vetor com 3 elementos do tipo da struct de pessoa que você criou. Você deve solicitar os dados de 3 pessoas para o usuário, e cadastrar nesse vetor. Antes de cadastrar, você deve verificar se o cpf é válido. Se o cpf for inválido, informe ao usuário, e solicite para que ele digite outro.
  - d. Depois de preencher o vetor com as 3 pessoas, você deve imprimir os dados na tela.

# Licença

Esta obra está licenciada com uma Licença [Creative Commons Atribuição 4.0 Internacional](https://creativecommons.org/licenses/by/4.0/).