

# Apresentação da disciplina

Prof. Carlos A. Astudillo

✉ [castudillo@ic.unicamp.br](mailto:castudillo@ic.unicamp.br)

📖 Sistemas Operacionais (MC504A)

📅 2º Semestre 2023

# Sobre o Professor

## ■ Colômbia



# Sobre o Professor



- Colômbia
- B.Eng. Electronics and Telecomm., UNICAUCA, CO

# Sobre o Professor



- Colômbia
- B.Eng. Electronics and Telecomm., UNICAUCA, CO
- M.Sc. Computer Science, UNICAMP

# Sobre o Professor



- Colômbia
- B.Eng. Electronics and Telecomm., UNICAUCA, CO
- M.Sc. Computer Science, UNICAMP
  - ▶ University of Trento, IT

# Sobre o Professor



- Colômbia
- B.Eng. Electronics and Telecomm., UNICAUCA, CO
- M.Sc. Computer Science, UNICAMP
  - ▶ University of Trento, IT
- Ph.D. Computer Science, UNICAMP, BR

# Sobre o Professor



- Colômbia
- B.Eng. Electronics and Telecomm., UNICAUCA, CO
- M.Sc. Computer Science, UNICAMP
  - ▶ University of Trento, IT
- Ph.D. Computer Science, UNICAMP, BR
  - ▶ University of British Columbia (UBC), CA

# Sobre o Professor



- Colômbia
- B.Eng. Electronics and Telecomm., UNICAUCA, CO
- M.Sc. Computer Science, UNICAMP
  - ▶ University of Trento, IT
- Ph.D. Computer Science, UNICAMP, BR
  - ▶ University of British Columbia (UBC), CA
  - ▶ University of Manitoba (UM), CA



- MC504A

# Sobre o Professor



- Colômbia
- B.Eng. Electronics and Telecomm., UNICAUCA, CO
- M.Sc. Computer Science, UNICAMP
  - ▶ University of Trento, IT
- Ph.D. Computer Science, UNICAMP, BR
  - ▶ University of British Columbia (UBC), CA
  - ▶ University of Manitoba (UM), CA
- Asst. Professor, IGCE/UNESP
- Asst. Professor, IC/UNICAMP

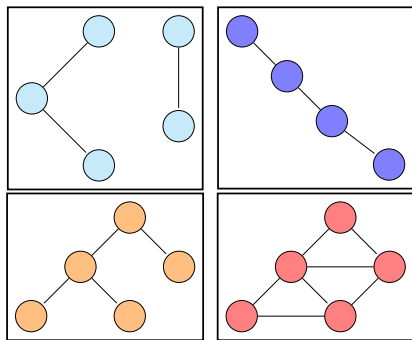
# Sobre o Professor (Interesse)

- 5G/6G
- Machine Learning Applied to Communications and Networking
- IoT and Cellular IoT
- Network Management
- Wireless networks
- Optical networks
- Mobile fronthauling and backhauling
- High-Performance Computing
- Deep Learning applied to Seismic Data
- Network protocols
- Edge Computing

# O que é aprender?

# O que é aprender?

- Processo cognitivo que **exige trabalho**
- Criar um conflito cognitivo e resolvê-lo
- Acomodando estruturas mentais
- Distintos tipos de conhecimento
  - ▶ O que procurar?
  - ▶ Como evolui-lo?



# Qual modo de aprender você usa?

## Baseada no ensino

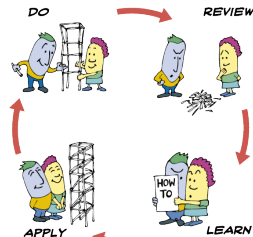
- O professor tem que me ensinar
  - ▶ Os conteúdos são passados
  - ▶ Os conteúdos são o centro
- Focado nos **conteúdos** (e repeti-los)



Waiting for superman. <https://vimeo.com/19329550>  
MC504A

## Baseada no aprendizado

- Eu tenho que aprender
  - ▶ Busca-se o entendimento
  - ▶ Várias maneiras de aprender
- Aprendizado ativo, e focado em **entender**



<https://asmaaalnahrawy.wordpress.com/2015/05/29/active-learning/>

# Nossa abordagem

- Nós usaremos um paradigma baseado no aprendizado
- Procuraremos um aprendizado ativo
  - ▶ Atividades
  - ▶ Discussões
  - ▶ Trabalhos para promover os conceitos
- Desenvolveremos pensamento crítico
- Atividades baseadas em resultados (com limite de tempo para as entregas)
  - ▶ A teoria é **importante**,
  - ▶ e **prática correta** também
- Se mudarmos o paradigma, temos que mudar nossa forma de pensar e agir

# Dados $\Rightarrow$ Sabedoria

■ Informação =  $\square \neq \square$  Conhecimento



# Dados $\Rightarrow$ Sabedoria

- Informação =  $\square \neq \checkmark$  Conhecimento
- Como o conhecimento é criado?
  - ▶ Absorver dados
  - ▶ Compreender a informação
  - ▶ Inferir a partir da informação

# Dados $\Rightarrow$ Sabedoria

- Informação =  $\square \neq \checkmark$  Conhecimento
- Como o conhecimento é criado?
  - ▶ Absorver dados
  - ▶ Compreender a informação
  - ▶ Inferir a partir da informação
- Aplicar os conhecimentos
  - ▶ Como aplicar algo que você não tem?
  - ▶ Como consertar algo que você não entende?

# É um processo

- Ler para entender a informação
- Processar (individual e colectivamente) para gerar conhecimento
  - ▶ Leva tempo
  - ▶ Precisa amadurecer
- A disciplina funciona se todos estamos na mesma página
  - ▶ Ler
  - ▶ Discutir e analisar
  - ▶ Tomar notas!
  - ▶ Programar e desenvolver
  - ▶ Realimentar
- É um processo

# Regime geral

- SO é uma disciplina densa
- Muito conteúdo, conceitos “novos”
  - ▶ Abstração do kernel
  - ▶ Interface de programação
  - ▶ Concorrência e threads
  - ▶ Sincronização de objetos compartilhados
  - ▶ Escalonamento
  - ▶ Endereços de memória
  - ▶ Memória virtual
  - ▶ Administração da memória
  - ▶ Armazenamento

Q: É muito conteúdo para um semestre só?

A: Não. Mas tem que aprender a administrar seu tempo

💡: Muitas coisas novas para aprender!

# Hábitos de estudo e trabalho

- Desenvolver uma estratégia de estudo logo no início
  - ▶ Devem ler
  - ▶ Devem fazer (programar/desenvolver)
  - ▶ Tomar nota (traduzir SO ao português)
  - ▶ Boa leitura: [https://reddit.com/comments/2b0yf8/good\\_students\\_how\\_do\\_you\\_go\\_about\\_getting\\_good/cj0qre2/](https://reddit.com/comments/2b0yf8/good_students_how_do_you_go_about_getting_good/cj0qre2/)
- Desenvolver uma dinâmica de trabalho em grupo
- Definir um cronograma para ler, programar, estudar
- **A disciplina é densa e trabalhosa** (prático e teórico), se não começarem agora, não dará tempo de

# Horas de trabalho

- SO: ~150 horas (semestrais)
    - ▶ 1 semestre = ~15 semanas
    - ▶ 1 semana = ~10 horas
  - 4 horas na sala de aula, **10 horas de estudo/trabalho independente** (semanais)
  - O trabalho está projetado considerando esse tempo para seu trabalho
  - ⌚ Lembrem que os grupos somam horas de trabalho
    - ▶ 2 pessoas: 20 horas semanais
    - ▶ 3 pessoas: 30 horas semanais
    - ▶ ...
- mas tem que considerar a sobrecarga da **administração**

# Livro de texto

- **Operating Systems: Principles & Practice**, Adnerson & Dahlin
  - ▶ Una visão mais moderna de sistemas operacionais
- (Tradicional) **Operating System Concepts**, Silberschatz, Galvin, & Gagne
  - ▶ Este também tem os tópicos, cobre mais segurança, visão “mais antiga”
  - ▶ Podem usá-lo também, mas têm que revisar a concordância dos temas com OP: PP
- Teremos **leituras obrigatorias** (semanais)
  - ▶ **Compreensões de leitura para aula**
  - ▶ Calendário no website da disciplina
- Muitos conceitos tomados do livro
- Encontrem o livro **agora!**

# Objetivos

- Sistemas operacionais
  - ▶ O que é?
  - ▶ Decisões de projeto (*design*)
  - ▶ Construção
- Programação em dupla
  - ▶ Projeto, documentação
  - ▶ Controle de versões
  - ▶ Habilidades interpessoais
- Apresentação de informação técnica
  - ▶ Escrita



# Aulas

- Muitos tópicos serão cobertos através do texto (leiam!)
- Não assistir as aulas afetará sua nota
  - ▶ O mapa não é o terreno, **os slides não são a aula**
  - ▶ Se falta, perde as perguntas e respostas (isto é um processo. . . )
  - ▶ Teremos atividades na aula (algumas com nota, tem que assistir para não perdê-las)
- Eu espero que vocês assistam as aulas
  - ▶ Lembrem do paradigma de aprendizado?
  - 🕒 Para cimentar o conhecimento e entender, temos que conversar e discutir os temas

# Atividades

## Projetos



### Projetos (tentativos):

- ▶ Exploração do ferramenta
- ▶ Gerenciamento de processos
- ▶ Gerenciamento de threads com semáforos
- ▶ Escalonamento de Processos



### Ambiente de desenvolvimento:

- ▶ Máquinas virtuais no computador
- ▶ SO: Minix
- ▶ Linguaje: C
- ▶ Controle de versões: `git`
- ▶ Documentação do código: `doxygen` (no possível)

# Disciplina

- Estamos seguindo uma disciplina do CMU
- Atividade e discussões de nível internacional
- Aproveitar nosso tempo
- Pouco tempo no semestre

# Calendário

## Tentativo

- 30 sessões de aula (incluindo provas)
- Revisão de conteúdo periódico
- Trabalhos (tarefas, compressões de leitura) quase toda semana.
- 3-4 projetos (a cada 7-8 aulas aprox.)

# Programação em equipe

## ■ Por que?

- ▶ Ajuda a resolver problemas maiores e complexos
- ▶ Ensina **habilidades interpessoais** necessárias no trabalho
  - Estabelecer marcos (milestones)
  - Estabelecer um fluxo de trabalho produtivo
  - Administrar seu tempo (antes que seja muito tarde)

## ■ Programação em equipe $\neq$ *engenharia de software*

- ▶ Não tem análise de requerimentos
- ▶ Não tem releases, projeto para seguir, ...
- ▶ Não é um ciclo de desenvolvimento completo

# Problemas com seus colegas

- **Alguem** terá problemas com seus colegas na equipe
  - ▶ Precisam se organizar cedo
    - A maioria dos problemas tem solução
    - Senão, poderemos reduzir o dano . . . só se temos tempo
  - ▶ **Não deixem os problemas de lado**
    - Se falarem no final do semestre, não posso ajudar

# Avaliações

- Leituras semanais, avaliadas na aula
  - ▶ Perguntas e respostas
  - ▶ Atividades na sala
  - ▶ **Compressões de leitura** (datas no calendário no website da disciplina)
- Relatórios
  - ▶ Todos os projetos tem um relatório no final
- Projetos
  - ▶ Lembram?
  - ▶ 2-3 grandes e um pequeno

# Esclarecimentos

- Os relatórios são entregues **só** no website da disciplina e **na data indicada**
- Os relatórios devem ser escritos usando  $\text{\LaTeX}$  usando o template **IEEETrans**
  - ▶ Entrega do documento (.pdf) e código fonte (.tex)
  - Q: Não conheço  $\text{\LaTeX}$ , o que faço?
  - A: Comece a aprender hoje (tem muitos **recursos online**)
- **Não serão aceitas entregas atrasadas** (tolerância zero)



# Ponderação

## ■ Média parcial

- ▶ Provas 1/2 (2/5 P1 e 3/5 P2)
- ▶ Projetos 3/8
- ▶ Trabalhos<sup>1</sup> 1/8

## ■ Média final

- ▶ Se tem média parcial inferior a 5 dá exame final
- ▶ Senão, é a média parcial

# Projetos

- Cada projeto está dividido em:
  - ▶ 70% desenvolvimento prático (atingir dos objetivos, desenvolvimento de software, explicação das decisões, completitude, etc.)
  - ▶ 30% relatório

# Relatórios

- Avaliação do relatório:
  - ▶ Introdução ao problema 15%
  - ▶ Resolução do problema 60%
  - ▶ Conclusões e discussões 15%
  - ▶ Aspectos formais 10%
- Explicar e detalhar o trabalho feito no relatório de forma sucinta (**máximo 4 páginas no relatório**)

# Horários

## ■ Aulas:

- ▶ Segundas feiras das 19h às 21h e quartas feiras das 21h às 23h na sala CB10

## ■ Avaliações

- ▶ Prova 1: 4 de outubro
- ▶ Prova 2: 29 de novembro
- ▶ Exame: 11 de dezembro

## ■ Atendimento:

- ▶ Email: [castudillo@ic.unicamp.br](mailto:castudillo@ic.unicamp.br) (só se for problema particular)
- ▶ Dúvidas: fórum do website da disciplina

# Website da disciplina

- Google Classroom: <https://classroom.google.com>
- Disciplina: G\_MC504A\_2023S2
- Gitlab: <https://gitlab.unicamp.br/>
- Material, tarefas, relatórios serão entregues no classroom, e código no gitlab