

# Roteiro de aulas

Serão 15 semanas de aulas, contendo:

- 2 provas
- 2 aulas de dúvidas
- 11 aulas de conteúdo

Os slides a seguir apresentam uma definição preliminar de tópicos que ainda precisam ser agrupados para compor as aulas.

# Aula 1

- Apresentação da disciplina com regras e avaliação
- Introdução super básica à sintaxe assembly e programação básica

# Aula 2

- Complementação da sintaxe assembly com mais instruções
- Até aqui o aluno já deve conhecer as principais instruções de montagem e ser capaz de escrever programas simples
- Principais estruturas de execução de código (if, while, for, etc)

# Aula 3

- Sistemas de numeração
- Representação de números em binário
- Representação de números em hexadecimal
- Representação de números em complemento de 2
- Operações com números binários
- Endianess
- Representação de caracteres
- Representação de strings

# Aula 4

- Organização básica de computadores
- Ciclo de instrução
- Registradores
- Memória
- Periféricos
- Endereçamento
- Ciclo de desenvolvimento de código

# Aula 5

- Tipos de instruções
- Codificação de instruções
- Decodificações de instruções
- Extensões RISC-V

# Aula 6

- Representação de números em ponto flutuante
- Instruções de ponto flutuante de precisão simples e dupla

# Aula 7

- Convenção de chamada de registradores
- Convenção de chamada de subrotinas
- ABI



# Aula 8

- Endereçamento de memória
- Organização de memória
- Pilha
- Heap
- Variáveis globais e locais
- Estruturas de dados e memória
- Estruturas dinâmicas

# Aula 9

- Exceções e interrupções
- Registradores especiais
- Convenções e alternativas de tratamento de interrupções

# Aula 10

- Entrada e saída
- Dispositivos de entrada e saída
- Chamadas de serviços do sistema operacional

# Aula 11

- Outros processadores e suas linguagens de montagem
- Intel x86
- ARM

# Aula 12

- Integração de código assembly com código em outras linguagens