

# MC504 - Sistemas Operacionais

Primeiro Semestre de 2014

[Islene Calciolari Garcia](#)

## Projeto 1: Sudoku Multithread

### Alunos:

136640 - Lucas Henrique Moraes

139546 - João Pedro Ramos Lopes

### Arquivo makefile:

Entradas:

main.c, main.h, analise.c, analise.h, solver.c, solver.h

Saída:

main

### Comandos:

lerM                      Le sudoku. A seguir, e necessario digitar 81 inteiros.

lerA                      Le sudoku de arquivo. A seguir, e necessario digitar o endereco do arquivo que contenha o sudoku

imprimirSudoku        Imprime o sudoku que foi lido pelo comando 'ler'

ajuda                    Imprime esse menu de ajuda na tela

verificar               Imprime todos os erros, se presentes, no sudoku

dica                     Imprime dicas para posições marcadas com 'X' no sudoku

resolver                Imprime resolução de sudoku lido. Deve ser válido

limpar                  Apaga sudoku lido

sair                      Sai do programa

### Módulo para verificação de diagramas completos

- Ler matriz de sudoku (comando 'lerM' ou 'lerA') e então utilizar 'verificar'.

### Módulo para a exibição de dicas

- Ler matriz de sudoku (comando 'lerM' ou 'lerA') e então utilizar 'dica'.

### Módulo para a exibição da solução completa

- Ler matriz de sudoku (comando 'lerM' ou 'lerA') e então utilizar 'resolver'.

**Estratégia adotada:**

A estratégia adotada foca na velocidade de resolução de cada módulo. No caso da solução completa, temos um sistema com 100% de soluções corretas e, mesmo no sudoku considerado mais difícil, em que são testadas quase 3 milhões de possibilidades e mais ou menos 8 mil chamadas a threads, tem-se uma resolução que não extrapola a escala dos décimos de segundos. Como ponto negativo, tem-se o crescimento linear do gasto de memória de acordo com o número de threads criadas, visto que cada thread tem uma cópia da estrutura de dados necessária para a resolução do problema.

**Verifica:**

Cada linha, coluna e quadrado possui uma thread de verificação da integridade do sudoku de acordo com as regras padrões.

**Dica:**

Cada linha, coluna e quadrado possui uma thread de verificação de possibilidades. A seguir, uma posição que pede dica ('X') compara os dados da linha, da coluna e do quadrado aos quais pertence e apresenta as possibilidades para a sua posição.

**Resolver sudoku:**

Limitando o máximo de threads a 8 (valor definido no código, mas facilmente alterável), sempre que possível um backtracking cria uma nova thread para analisar um determinado galho de possibilidades. Caso não haja threads disponíveis, ela aprofunda no dado galho.