

”

E-fólio A | Folha de resolução para E-fólio

UNIDADE CURRICULAR: Sistemas Computacionais

CÓDIGO: 21174

DOCENTE: Nelson Russo

A preencher pelo estudante

NOME: Elen Fernandes de Oliveira

N.º DE ESTUDANTE: 2404716

CURSO: Licenciatura em Engenharia Informática

DATA DE ENTREGA: 08 de novembro de 2025

- 1) A Lei de Moore afirmava que o número de transístores num chip duplicaria a cada 18 a 24 meses, permitindo aumentar o desempenho e reduzir os custos de forma contínua. No entanto, este crescimento acabou por abrandar devido a limitações físicas, como o aquecimento, a dissipação de calor e o aumento do consumo energético, que tornaram impossível continuar a miniaturização indefinidamente. Para ultrapassar esses limites e manter a evolução do desempenho, surgiram novas arquiteturas que exploram o paralelismo, o *pipelining* e a hierarquia de memórias de formas inovadoras.

As arquiteturas heterogéneas (CPU + GPU) combinam processadores com funções distintas. O CPU é responsável por tarefas de controlo e execução sequencial, enquanto o GPU é otimizado para processar milhares de dados em paralelo. Desta forma, aproveita-se o paralelismo de instrução de dados, aumentando o desempenho sem depender apenas da frequência do relógio.

A computação neuromórfica inspira-se na estrutura e funcionamento do cérebro humano, onde milhares de “neurónios artificiais” comunicam de forma paralela e assíncrona. Este tipo de arquiteturas processa informações de modo distribuído e com baixo consumo energético, sendo especialmente eficaz em tarefas cognitivas e de aprendizagem. Aqui, a hierarquia de memórias é repensada, aproximando os dados do local de processamento e reduzindo a latência.

A computação quântica representa uma rutura com o paradigma clássico binário. Baseia-se em qubits, que podem representar múltiplos estados em simultâneo (superposição) e interagir entre si (entrelaçamento), permitindo um paralelismo exponencial em certos tipos de problemas. Apesar dos desafios de estabilidade e correção de erros, esta abordagem promete ganhos significativos em áreas específicas como criptografia e otimização.

Em todas estas arquiteturas, o *pipelining* é utilizado para sobrepor etapas de execução, aumentando o *throughput*, enquanto a hierarquia de memórias é otimizada com o uso de caches e memórias locais para minimizar a latência.

Assim, mesmo com o abrandamento da Lei de Moore, o desempenho continua a crescer graças ao paralelismo, à especialização do hardware e à inovação arquitetónica, demonstrando que o futuro da computação depende menos da miniaturização e mais da inteligência no desenho das suas estruturas.

Fonte: RF 1.1 (Petterson & Hennessy, 2017)

Usei a IA-G para rever a clareza e coerência da explicação, garantindo que cada conceito técnico fosse aplicado corretamente às arquiteturas referidas.

- 2) a) O programa executa em 120 segundos, sendo 45% paralelizável. Com 4 processadores e *overhead* desprezável, o tempo total é calculado pela Lei de Amdahl através da fórmula:

$$T_{\text{total}} = T * \left((1 - P) + \frac{P}{N} \right)$$

Sendo que T é o tempo total inicial em segundos, P é a parte paralelizável e N é o número de processadores. Neste contexto:

$$T = 120$$

$$P = 45\% = 0,45$$

$$N = 4$$

$$T_{\text{total}} = 120 * \left((1 - 0,45) + \frac{0,45}{4} \right) = 120 * (0,55 + 0,1125) = 120 * 0,6625 = 79,5s$$

Tempo total = 79,5 segundos

- b) Aplicando o overhead de 10%:

$$\begin{aligned} T_{\text{total}} &= 120 * \left((1 - 0,45) + \frac{0,45}{4} * 1,10 \right) = 120 * (0,55 + 0,12375) \\ &= 120 * 0,67375 = 80,85s \end{aligned}$$

Tempo total = 80,85 segundos

- c) Análise:

O *overhead* reduziu a eficiência do paralelismo, aumentando o tempo de 79,5 segundos para 80,85 segundos. Segundo a Lei de Amdahl, mesmo com mais processadores, o ganho de desempenho é limitado pela fração sequencial do programa e pelos custos de coordenação entre as tarefas. Assim, mesmo com processadores infinitos, o tempo mínimo seria $120 * 0,55 = 66$ segundos. O paralelismo melhora o desempenho até certo ponto, mas o *overhead* de comunicação e sincronização reduz os ganhos práticos, mostrando a importância de minimizar essas perdas.

Fontes: RF 1.4 e 1.5 – Desempenho Computacional e RF 1.6 – aula 2: Medida de Desempenho

Usei a IA-G para confirmar a coerência dos cálculos da Lei de Amdahl e rever a clareza da explicação em cada alínea.

3)

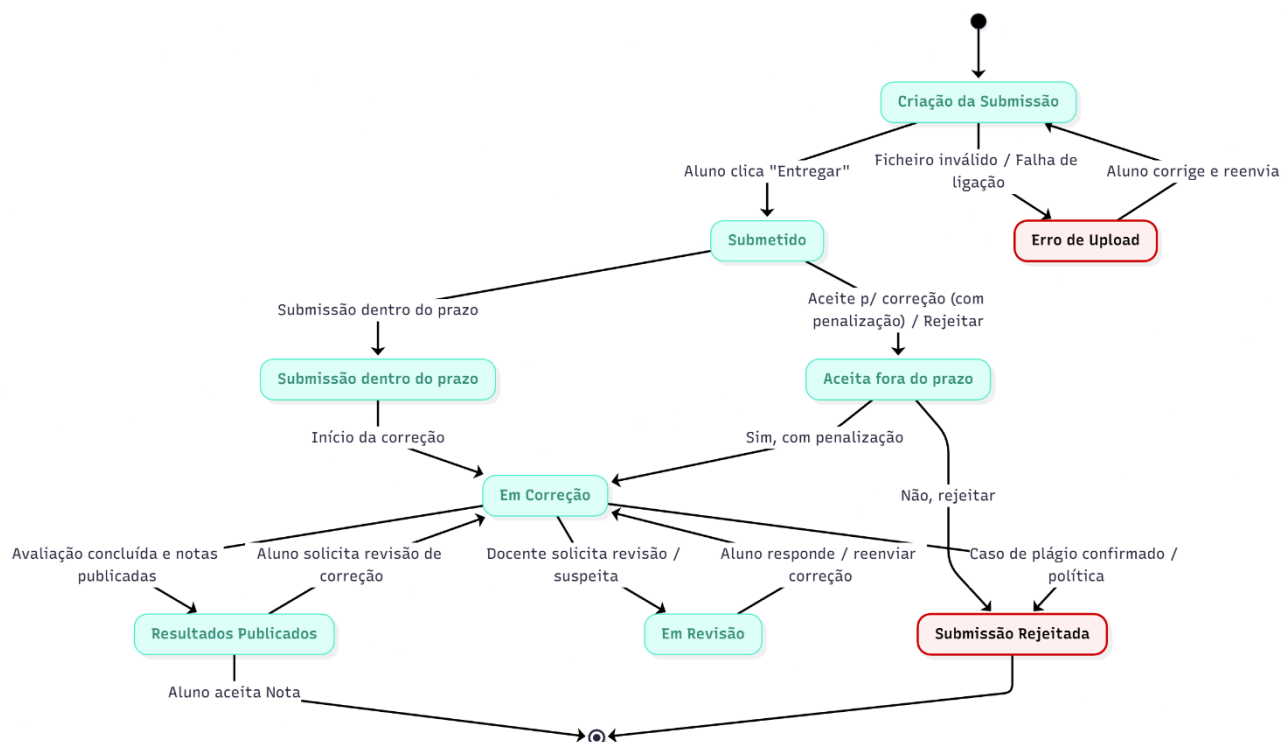


Diagrama 1 - Máquina de Estados da Submissão de um E-Fólio

Modelei o processo como uma máquina de estados do tipo Moore, em que o comportamento do sistema (ex.: *Em Correção* e *Resultados Publicados*) depende apenas do estado atual e mantém-se enquanto este mantiver ativo.

O modelo é sequencial e determinístico, apresentando um único estado ativo de cada vez e transições únicas para cada evento.

Incluí duas situações de erro (*Erro de Upload* e *Submissão Rejeitada*), com caminhos definidos de resolução ou rejeição.

O diagrama foi criado em Mermaid Chart, ferramenta que permite gerar diagramas de estados padronizados, compatíveis com a notação ensinada na unidade curricular.

Fonte: RF 2.0 a 2.6 – Estados e Máquinas de Estados

Recorri à IA-G para validar a coerência dos estados e das transições, assegurando o cumprimento dos critérios do enunciado.