

Lista de Exercícios

Revisão de Arquiteturas e Organização de Computadores

Segundo Semestre de 2020

- Os processadores de algumas décadas atrás tinham uma organização bem mais simples do que os processadores correntes.
 - O que possibilitou o aumento de complexidade nos processadores?
 - Com qual objetivo foi implementado esse aumento de complexidade?
- Comente sobre a diferença entre as melhoras em tempo de execução de operações pelas CPUs e tempo de acesso à memória com o desenvolvimento de novas tecnologias.
- Qual a função de um barramento num computador? Qual a importância da sua largura de banda?
- Considerando a hierarquia de memória presente em computadores atuais:
 - No que ela consiste?
 - Por que ela é necessária?
- Sobre localidade em programas:
 - O que é localidade temporal?
 - O que é localidade espacial?
- No trecho de código delimitado abaixo, indique onde existe localidade temporal e onde existe localidade espacial.

```
double s = 0;
for (int i = 0; i < N; ++i) {
    s += a[i] * b[i]
}
```
- O trecho de código abaixo tem localidade pobre. Explique a razão.

```
for (int i = 0; i < N; ++i) {
    a[100*i] = 1;
}
```
- O que é uma memória cache?
- Qual a relação entre caches e localidade?
- Qual a relação entre uma linha de cache e um bloco de memória?
- Por que razão os dados são transportados para a memória cache em blocos, ao invés de individualmente em bytes?

12. Considere um sistema onde o endereçamento é de 32 bits, ele dispõe de uma memória cache com 8 Mbytes que usa linhas de 32 bytes e mapeamento associativo por conjuntos de 2 linhas. Indique quantos bits do endereço são usados para offset, quantos para o identificador do conjunto e quantos para o rótulo do bloco.
13. Um programa específico realiza um total de $2 \cdot 10^{10}$ acessos à memória. Suponha um sistema computacional onde o tempo de acesso de um *cache hit* é de $0.5 \cdot 10^{-9}s$ e de um *cache miss* é de $100 \cdot 10^{-9}s$. Qual será o tempo gasto com acessos à memória por esse programa se o *hit ratio* for de 0.8? E se o *hit ratio* for de 0.99?
14. Considerando que um processo em execução pode estar nos estados **pronto**, **executando** ou **bloqueado**, indique quais transições que podem ocorrer entre esses estados e quais os eventos que as provocam.
15. Explique por que é necessário o particionamento da memória em sistemas de multiprogramação.
16. Num sistema de memória virtual, qual a diferença entre endereço físico e endereço virtual?
17. O que é uma *página* em um sistema de memória virtual? Qual sua relação com *quadros de página*?
18. Qual a função da tabela de páginas em um sistema de memória virtual?
19. Endereços virtual consecutivos não precisam necessariamente estar em endereços físicos consecutivos. Por que isso é possível? Qual a vantagem? Existe possível impacto com relação a localidade?
20. Qual é o problema de *fragmentação de memória* e por que ele é resolvido em sistemas de memória virtual.
21. Como um sistema de memória virtual lida com o problema de alocação e liberação dinâmicas de memória durante a execução de um programa?
22. O que é um *page fault* e qual a sua relação com paginação sob demanda?
23. Qual a função do *swap* num sistema de memória virtual?
24. Explique no que consiste o problema denominado *trashing* em memórias cache e paginação.
25. O que é uma tabela invertida de páginas? Por que ela é necessária, ao invés de uma implementação direta da tabela de páginas?
26. O que é o *translation lookaside buffer* (TLB) e por que ele é necessário?
27. O que é um *pipeline* de instruções e qual seu objetivo? Quais suas limitações?
28. O que é um sistema de previsão de desvios? Qual sua relação com *pipelines*?
29. Quando dizemos que uma CPU é *superescalar*? Qual a relação com o termo *instruction level parallelism* (ILP)?
30. O que são dependências de dados, e por que elas são importantes em CPUs superescalares?
31. Descreva o que caracteriza os seguintes tipos de dependência:
 - (a) Dependência verdadeira (ou dependência de dados).
 - (b) Dependência procedural.
 - (c) Dependência de saída.

(d) Antidependência.

32. Por que razão apenas a dependência de dados é considerada *verdadeira*?

33. O que é *renomeação de registradores*? Qual sua relação com execução superescalar?