Programação e Desenvolvimento de Softwares II

André Felipe de Assis Rodrigues 2024012099

Semana 2

**Exercícios Objetivos**

1. Alternativa E. Um ponteiro é um tipo de variável que armazena um endereço de memória para determinado tipo de dado, seja um *char*, *int*, *float*, *double*, etc. Desse modo, um ponteiro só consegue apontar para o tipo de dado para o qual ele foi declarado, justificando a alternativa (a), e apenas um ponteiro do tipo *void* conseguiria apontar para qualquer tipo de dado, uma vez que ele é um ponteiro genérico para um *byte*. Um endereço de memória é um código em hexadecimal que armazena as variáveis e funções do código fonte, e seu tamanho é limitado pelo *size\_t* da arquitetura do computador, majoritariamente 32 ou 64 bits.
2. Alternativa D. O modelo da hierarquia de memória é um diagrama utilizado para demonstrar a relação diretamente proporcional entre a capacidade de armazenamento e o tempo de acesso em arquiteturas de memória. Seus componentes são estruturados com base na frequência em que certos dados são requisitados. Dessa maneira, as memórias em disco costumam ocupar a base de tal modelo, pois são extensas em volume, e portanto possuem alto tempo para localizar determinado dado. De modo contrário, os registradores compõem a parte mais velozmente acessível da RAM, pois como seu volume é relativamente menor, a capacidade para achar determinados dados é maior.
3. Alternativa C. O overflow ocorre quando determinado espaço para um tipo de dado é insuficiente para os operações realizadas no programa, assim, quando seu limite é extrapolado, comportamento não-definido pode ser apresentado e gerar bugs, em C e C++, o comportamento implementado é o *wrap*, que categoriza o reinício da contagem dos *bits*. Logo, as outras alternativas estão erradas por se tratarem de outros conceitos, como por exemplo o *buffer* é uma memória temporária, de relativamente pouco volume, usada para o trânsito eficiente dos dados. Outro exemplo é o *bit* de *carry* que é utilizado na aritmética binária, especialmente de complemento de dois, categorizado pelo bit mais à esquerda de um dado, que facilita as transferências de “vai um”, “vem um” entre os *bits*.
4. Alternativa E. O ciclo de memória é o intervalo usado pelo CPU para acessar a memória e realizar a operação de escrita ou leitura, que pode variar conforme o *hardware* da memória e é rotineiramente associado à performance do sistema. Dentre outros conceitos apresentados na questão, a volatilidade se refere à persistência de determinado dado na memória após o desligamento do sistema, podendo ser perdida no caso da RAM, ou mantida no caso de discos rígidos, e a capacidade se refere ao volume da unidade de armazenamento e à quantidade de dados que podem ser escritos nela.
5. Alternativa C. O processo de alocação e desalocação de memória é parte do gerenciamento da memória de um sistema, e atribui um endereço de memória representado em hexadecimal para determinado bloco da memória. Os processos de alocação podem ser estáticos ou dinâmicos, os estáticos ocorrem no início do programa, garantindo que haja espaço suficiente para todas as variáveis e funções utilizadas no programa, enquanto os dinâmicos alocam e desalocam memória no decorrer do programa, e podem causar latências e até congelamentos, caso o sistema não tenha memória suficiente para as necessidades do programa. A memória virtual é um modo do computador prover as necessidades de memória de um programa, fazendo com que parte de sua execução seja carregada na memória e outra parte permaneça em disco, de modo que, caso seja necessário, a transferência de dados entre tais localidades ocorre por meio da po processo de paginação.