

2. Respostas sobre alocação dinâmica de memória

a) **Comportamento da função free:** A função free libera a memória que foi alocada dinamicamente. Após liberar a memória, o ponteiro não deve mais ser acessado.

b) **Após chamar free, o ponteiro pode ser utilizado?** Não. Após chamar free, o ponteiro se torna inválido e seu acesso pode causar comportamento indefinido.

c) **O que causa vazamentos de memória?** Vazamentos de memória ocorrem quando a memória alocada dinamicamente não é liberada com free. Isso ocorre se o ponteiro que aponta para a memória for perdido antes de chamar free.

d) **O que a instrução malloc retorna quando não consegue realizar a alocação?** Quando malloc não consegue alocar memória, ele retorna NULL.

e) **Explicação da instrução calloc:** calloc aloca memória para um número especificado de elementos e inicializa essa memória com zero.

f) **Diferença entre malloc e calloc:** malloc aloca um bloco de memória de tamanho especificado, mas não inicializa o conteúdo. calloc, além de alocar, inicializa a memória com zero.

3. Manipulação de Arquivos

a) **Modos de abertura de arquivos:**

- "r": Abre um arquivo para leitura.
- "w": Cria um novo arquivo para escrita (ou sobrescreve um existente).

- "a": Abre um arquivo para adicionar conteúdo no final (não sobrescreve).
- "rb", "wb", "ab": Modos binários, usados para arquivos não-texto.
- "r+", "w+", "a+": Modos para leitura e escrita.

b) Funções fgetc, fprintf, fread, fwrite:

- fgetc: Lê uma linha de um arquivo, incluindo espaços, até encontrar uma nova linha ou o fim do arquivo.
- fprintf: Escreve uma linha formatada em um arquivo, semelhante a printf.
- fread: Lê um bloco de dados de um arquivo.
- fwrite: Escreve um bloco de dados em um arquivo.

4. Saída do código

A saída do código será 0.45, pois:

- A condição $a > b \parallel !(a > 0)$ é verdadeira, então o cálculo será b / a , ou seja, $11 / 5 = 2$, e o valor será convertido para float resultando em 0.45.

5. Análise de variáveis

A análise das variáveis é feita com base nos cálculos realizados no código. Vamos verificar cada linha:

1. $a * b / c = 0.666$
2. $a * b \% c + 1 = 3$
3. $++a * b - c-- = 5$
4. $7 -- b * ++d = -13$

5. $a / b / c = 0.000$

6. $7 + c * --d / e = 8.700$

7. $2 * a \% - b + c + 1 = 7$

8. $39 / - ++e - + 29 \% \% c = -2.625$

9. $7 - + ++a \% \% (3+b) = 6$

6. Valor de w

O valor de w após a execução será 16, pois $y + z = 5 + 11 = 16$, mas a condição $y > z$ não é verdadeira, então o valor de w permanece como $y + z$.

8. O que é um vetor?

Um vetor em C é uma estrutura de dados que armazena elementos do mesmo tipo em posições consecutivas na memória. O vetor é tratado como um ponteiro, e os elementos podem ser acessados usando o índice do vetor, começando de 0. A alocação e manipulação de um vetor são feitas através de índices e os elementos podem ser acessados diretamente ou passados para funções.