

Descrição do projeto

Para este trabalho a sua equipe deverá desenvolver um programa em linguagem C capaz de avaliar expressões matemáticas escritas, com suporte a operações básicas matemáticas (adição, subtração, multiplicação, divisão, potenciação e radiciação) e funções especiais (seno e cosseno). Assim, o programa deverá ser capaz de converter e avaliar expressões entre as formas infixada e pós-fixada, utilizando **pilhas** como estrutura fundamental.

A **notação infixada** é o formato mais comum usado pelos seres humanos para escrever expressões matemáticas.

Nela, os operadores (como +, -, *, /) são escritos entre os operandos (números ou variáveis).

Por exemplo:

- $A + B$
- $3 * (4 + 5)$
- $(7 - 2) / 3$

Na notação infixada, a ordem correta das operações não depende apenas da sequência dos símbolos, mas também das regras de:

- Precedência dos operadores (ex.: multiplicação antes de soma);
- Associatividade (esquerda ou direita);
- Parênteses, que explicitam qual parte deve ser avaliada primeiro.

A **notação pós-fixada**, também conhecida como notação polonesa reversa (RPN - Reverse Polish Notation), é uma forma de escrever expressões matemáticas onde os operadores após seus operandos. Isso contrasta com a notação infixada, que é a forma mais comum e usa parênteses para indicar a ordem das operações.

As principais características da notação pós-fixada estão relacionadas à eliminação de parênteses e à forma simples de avaliação:

- Na notação pós-fixada, não há necessidade de parênteses para alterar a ordem das operações, pois a ordem dos operadores é suficiente para definir a prioridade. Assim, a expressão infixada $(3 + 4) * 5$ é expressa como $3 4 + 5 *$ na notação pós-fixada.
- As expressões pós-fixadas são avaliadas usando uma pilha, o que facilita a implementação em computadores e calculadoras. Logo, a avaliação é feita lendo a expressão da esquerda para a direita, empilhando operandos e aplicando operadores aos operandos no topo da pilha.

Por exemplo: para a expressão $2\ 3\ 4\ *\ +$ (notação pós-fixa), devem ser seguidos os seguintes passos, para a obtenção do resultado.

Símbolo lido	Ação	Pilha
2	Empilha 2	[2]
3	Empilha 3	[2,3]
4	Empilha 4	[2,3,4]
*	Desempilha 3 e 4 e calcula $3*4=12$ --> empilha o resultado	[2,12]
+	Desempilha 2 e 12 e calcula $2+12=14$ --> empilha o resultado	[14]

O seu projeto deverá:

I. Converter de infixada para pós-fixada:

- O programa deve receber uma expressão escrita pelo usuário no formato infixado (ex.: $(3 + 4) * 5$), processar a conversão para pós-fixada (ex.: $3\ 4\ +\ 5\ *$) e exibir o resultado.
- Essa conversão deve respeitar corretamente:
 - A precedência dos operadores ($*$ e $/$ antes de $+$ e $-$).
 - A associatividade (normalmente da esquerda para a direita).
 - O balanceamento de parênteses.

II. Converter de pós-fixada para infixada:

- O programa deve receber uma expressão escrita em formato pós-fixado (ex.: $3\ 4\ +\ 5\ *$), reconstruir a forma infixada (ex.: $(3 + 4) * 5$) e exibir para o usuário.
- Essa etapa exige usar pilhas para reconstruir a ordem dos operandos e inserir corretamente os parênteses que preservam a lógica da expressão original.

III. Realizar os testes envolvendo as operações de:

- Adição
- Subtração
- Multiplicação
- Potenciação
- Radiciação
- Seno
- Cosseno

O código deverá estar organizado no formato de um TAD, distribuído nos arquivos:

- `expressao.h`
- `expressao.c`
- `main.c`

Além disso, o programa deverá oferecer suporte a operações que envolvam o uso de números decimais (`float`).