

Avaliador de expressões numéricas

Este trabalho prático tem por objetivo desenvolver um programa em linguagem C capaz de avaliar expressões matemáticas escritas, com suporte a operações básicas e funções matemáticas especiais. O sistema também deverá ser capaz de converter e avaliar expressões entre as formas infixada e pós-fixada, utilizando **pilhas** como estrutura fundamental.

Além disso, o código-fonte deve:

- A) traduzir a expressão da notação infixada para notação pós-fixada;
- B) traduzir a expressão da notação pós-fixada para notação infixada;
- C) para operações com dois operandos, ser usados +, -, *, /, % e ^ para as operações matemáticas básicas;
- D) ser usados **raiz**, **sen**, **cos**, **tg** e **log** para raiz quadrada, seno, cosseno, tangente e logaritmo de base 10;
- E) os arcos a serem considerados para sen, cos e tg devem ser considerados em graus;
- F) as operações indicadas no item D devem ser aplicadas a somente um operando;
- G) avaliar expressões de forma a obter os valores constantes na seguinte tabela:

Teste	Notação Posfixa	Notação Infixa	Valor
1	3 4 + 5 *	(3 + 4) * 5	35
2	7 2 * 4 +	7 * 2 + 4	18
3	8 5 2 4 + * +	8 + (5 * (2 + 4))	38
4	6 2 / 3 + 4 *	(6 / 2 + 3) * 4	24
5	9 5 2 8 * 4 + * +	9 + (5 * (2 + 8 * 4))	109
6	2 3 + log 5 /	log(2 + 3) / 5	Aprox. 0.14
7	10 log 3 ^ 2 +	(log10)^3 + 2	3
8	45 60 + 30 cos *	(45 + 60) * cos(30)	Aprox. 90,93
9	0.5 45 sen 2 ^ +	sen(45) ^2 + 0,5	1

Outros testes poderão (e deverão) ser realizados.

O que deve ser feito

No desenvolvimento deste trabalho serão avaliados:

- Correção da solução proposta;
- Código-fonte disponibilizado e compartilhado no GitHub;
- Documentação por meio do template disponibilizado junto com o enunciado deste trabalho;
- Desenvolvimento do código-fonte de forma modularizada;
- Utilização da linguagem C padrão;
- Código-fonte distribuído nos arquivos expressao.c, expressao.h e main.c;
- Identificação de inconsistências nos dados de entradas; e
- Condições para que as operações sejam feitas.

Atenção:

O código-fonte deve ser estruturado em 3 (três) arquivos fonte, de nomes **expressao.c**, **expressao.h** e **main.c**, a serem compilados com **gcc expressao.c main.c -o expressao.exe**. Ademais, o arquivo de cabeçalho expressao.h é apresentado a seguir e não deve ser modificado:

```
#ifndef EXPRESSAO_H
#define EXPRESSAO_H

typedef struct {
    char posFixa[512];    // Expressão na forma pos-fixa, como 3 12 4 + *
    char inFixa[512];    // Expressão na forma infixa, como 3 * (12 + 4)
    float Valor;         // Valor numérico da expressão
} Expressao;

char *getFormaInFixa(char *Str); // Retorna a forma inFixa de Str (posFixa)
char *getFormaPosFixa(char *Str); // Retorna a forma posFixa de Str (inFixa)
float getValorPosFixa(char *StrPosFixa); // Calcula o valor de Str (na forma posFixa)
float getValorInFixa(char *StrInFixa); // Calcula o valor de Str (na forma inFixa)

#endif
```