

EXERCICIO 3

LETRA A

```
#include <stdio.h>

#include <stdlib.h>

#include <math.h>

float funcao(float x);

int main(void){

    // O intervalo inicial foi obtido através da análise do gráfico da função
    // Como a função possui 4 zeros, escolhi procurar aquele mais perto de (0,0)
    // Assim, o intervalo inicial de busca sera [-2,-1]

    float intervalo[2] = {-2, -1};

    float erro = abs(intervalo[1] - intervalo[0]);

    float x_k, fx_k, fa, fb;

    int iteracoes = 0;

    // Itera até que o erro seja menor que o desejado
    while (erro >= 0.00001) {

        // Atualiza os valores de x e f(x)
        x_k = (intervalo[1] + intervalo[0]) / 2;

        fx_k = funcao(x_k);

        fa = funcao(intervalo[0]);

        fb = funcao(intervalo[1]);

        // Atualiza o intervalo
        if (fx_k * fb < 0) {

            intervalo[0] = x_k;

        } else {

            intervalo[1] = x_k;

        }

        // Calcula o erro
        erro = fmin(fabs(fx_k), fabs(intervalo[1] - intervalo[0]));

        iteracoes++;

        // Prints da iteração
        printf("\nIteracao: %d\n", iteracoes);
```

```

printf("\tErro: %f\n", erro);
printf("\tx_%d: %f\n", iteracoes, x_k);
printf("\tf(x_%d): %f\n", iteracoes, fx_k);
}

// Resposta final
printf("\n\nA raiz foi encontrada apos %d iteracoes\n", iteracoes);
printf("O valor aproximado de x e: %f\n", x_k);
}

float funcao(float x) {
    return pow(x, 4) - pow(x, 3) - 30 * pow(x, 2) + 16 * x + 80;
}

```

LETRA B

```

#include <stdio.h>
#include <stdlib.h>
#include <math.h>

float funcao(float x);
float derivada(float x);

int main(void){
    float x_k = 4;
    float x_k1 = 4;
    float f;
    float erro = INFINITY;
    int iteracao = 0;

    // Itera até que o erro seja menor que o desejado
    while (erro >= 0.00001) {
        // Atualiza os valores de x e f(x)
        x_k = x_k1;
        f = funcao(x_k);
        x_k1 = x_k - f / derivada(x_k);

        // Atualiza o valor do erro
        erro = fmin(fabs(x_k1 - x_k), fabs(funcao(x_k1)));
    }
}

```

```

    iteracao++;

    // Prints da itera  o
    printf("\nIteracao: %d\n", iteracao);
    printf("\tErro: %f\n", erro);
    printf("\tx_%d: %f\n", iteracao, x_k1);
    printf("\tf(x_%d): %f\n", iteracao, funcao(x_k1));
}

// Resposta final
printf("\n\nA raiz foi encontrada apos %d iteracoes\n", iteracao);
printf("O valor aproximado de x e: %f\n", x_k1);
}

float funcao(float x) {
    return pow(x, 2) + logf(x) / 50;
}

float derivada(float x) {
    return 2 * x + 1 / (50 * x);
}

```

LETRA C

```

#include <stdio.h>
#include <math.h>

float funcao(float x);

int main(void) {
    // O intervalo inicial foi obtido atrav  s da an  lise do gr  fico da fun    o
    // Assim, os valores iniciais ser  o 2 e 3
    float x_k0, f_x_k0, f_x_k1, f_x_k2;
    float x_k1 = 2;
    float x_k2 = 3;
    float erro = INFINITY;
    int iteracoes = 0;
    while (erro >= 0.00001) {
        x_k0 = x_k1;

```

```

x_k1 = x_k2;
f_x_k0 = funcao(x_k0);
f_x_k1 = funcao(x_k1);
x_k2 = x_k1 - ((x_k1 - x_k0) / (f_x_k1 - f_x_k0)) * f_x_k1;
f_x_k2 = funcao(x_k2);
erro = fmin(fabs(f_x_k2), fabs(x_k2 - x_k1));
iteracoes++;
printf("\nIteracao: %d\n", iteracoes);
printf("\tErro: %f\n", erro);
printf("\tx_%d: %f\n", iteracoes, x_k2);
printf("\tf(x_%d): %f\n", iteracoes, f_x_k2);
}
}
float funcao(float x) {
    return sin(x) + cos(x);
}

```

EXERCICIO 4

LETRA C

```

#include <stdio.h>
#include <stdlib.h>
#include <math.h>
float funcao(float x);
float derivada(float x);
int main(void){
    float x_k = 4;
    float x_k1 = 4;
    float f;
    float erro = INFINITY;
    int iteracao = 0;
    // Itera até que o erro seja menor que o desejado
    while (erro >= 0.00001) {

```

```

// Atualiza os valores de x e f(x)
x_k = x_k1;
f = funcao(x_k);
x_k1 = x_k - f / derivada(x_k);

// Atualiza o valor do erro
erro = fmin(fabs(x_k1 - x_k), fabs(f));

iteracao++;

// Prints da iteração
printf("\nIteracao: %d\n", iteracao);
printf("\tErro: %f\n", erro);
printf("\tx_%d: %f\n", iteracao, x_k1);
printf("\tf(x_%d): %f\n", iteracao, f);
}

// Resposta final
printf("\n\nA raiz foi encontrada apos %d iteracoes\n", iteracao);
printf("O valor aproximado de x e: %f\n", x_k1);
}

float funcao(float x) {
    return pow(x, 2) - x - 1;
}

float derivada(float x) {
    return 2 * x - 1;
}

```