

Exercício 2 – ANN

Aluno: Lucas Meneghelli Pereira

Segue abaixo as imagens das resoluções das questões e os algoritmos implementados utilizando a linguagem de programação 'c' para o processamento dos métodos iterativos.

S T Q Q S S D

Ex 2 - ANN
- Lucas Meneghelli Pereira

1 - $p_1 = 1,474$ $f(x) = x^3 - 4x - 1$ $f'(x) = 3x^2 - 4$

$$p_{n+1} = p_n - \frac{f(p_n)}{f'(p_n)}$$
$$p_2 = 1,474 + \frac{3,693476}{2,518028} = 2,940813$$

doqui p/ frente meu algoritmo

$$p_3 = 2,363463$$
$$p_4 = 2,198039$$
$$p_5 = 2,115623 \rightarrow \text{resposta } f //$$

2 - $p_1 = -0,839$ $p_2 = 0,532$ $f(x) = x^3 - 4x - 1$

$$p_{n+1} = \frac{p_{n-1}(f(p_n)) - p_n(f(p_{n-1}))}{f(p_n) - f(p_{n-1})}$$
$$p_3 = \frac{-0,839 \cdot (-2,977437) - 0,532 \cdot 1,765410}{-2,977437 - 1,765410} = -0,928679$$

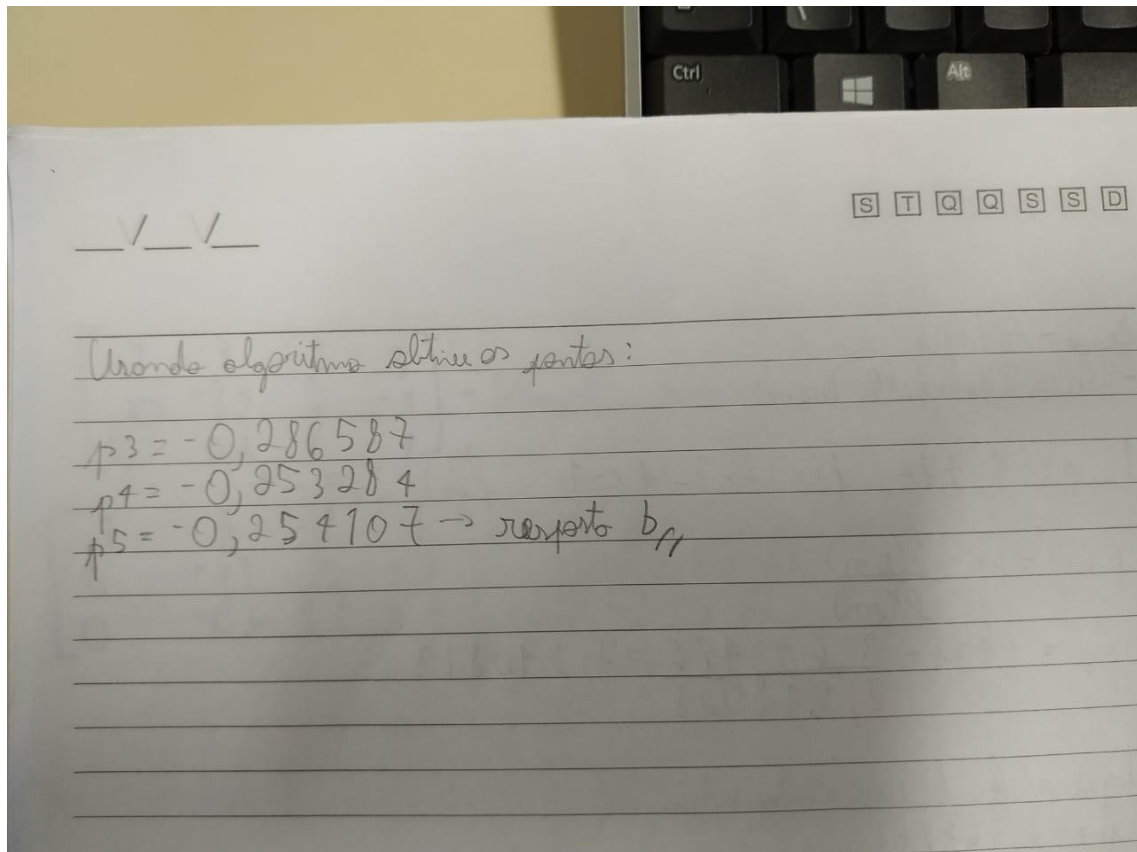
doqui p/ frente meu algoritmo

$$p_4 = -0,254888$$
$$p_5 = -0,254089 \rightarrow \text{resposta } d //$$

3 - $a_1 = -0,687$ $b_1 = 0,649$ $f(x) = x^3 - 4x - 1$

$$f(a_1) = 1,423757$$
$$f(b_1) = -3,308910$$

Segundo Bolzano, temos uma raiz entre a_1 e b_1 visto que $f(a_1) \cdot f(b_1) < 0$.



Algoritmo utilizado na questão 1:

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

int main() {

    float num, result, derivada, p;
    int i, n;

    scanf( "%f", &num );
    scanf( "%i", &n );

    for( i = 2; i <= n; i++ ){

        result = pow( num, 3 ) - ( 4 * num ) - 1;
        derivada = 3 * pow( num, 2 ) - 4;
        p = num - ( result / derivada );
        printf( "\n p%i: %f", i, p );
        num = p;

    }

    return 0;
}
```

Algoritmo utilizado na questão 2:

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

int main(){

    float n1, n2, f1, f2, result;
    int i, t;

    scanf( "%f", &n1 );
    scanf( "%f", &n2 );
    scanf( "%i", &t );

    for( i = 3; i <= t; i++ ){

        f1 = pow(n1,3) - (4 * n1) - 1;
        f2 = pow(n2,3) - (4 * n2) - 1;

        result = ((n1*f2)-(n2*f1))/(f2-f1);

        printf( "\np%i: %f", i, result );
        n1 = n2;
        n2 = result;

    }

    return 0;
}
```

Algoritmo utilizado na questão 3:

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

float calcFunc(float x){
    return pow(x, 3) - (4 * x) - 1;
}

int main(){

    float n1, n2, n3, f1, f2, f3;
    int i, t;

    scanf("%f", &n1);
    scanf("%f", &n2);
    scanf("%i", &t);

    f1 = calcFunc(n1);
    f2 = calcFunc(n2);

    if ((f1 * f2) < 0){

        for (i = 3; i <= t; i++){
            f1 = calcFunc(n1);
            f2 = calcFunc(n2);
            n3 = ((n1 * f2) - (n2 * f1)) / (f2 - f1);
            printf("\n%i: %f\n", i, n3);
            f3 = calcFunc(n3);

            if ((f2 * f3) < 0){
                n1 = n2;
                n2 = n3;
            }
            else{
                n2 = n3;
            }
        }

    }
    else{
        printf("Nao ha raizes entre os dois numeros.\n");
    }

    return 0;
}
```