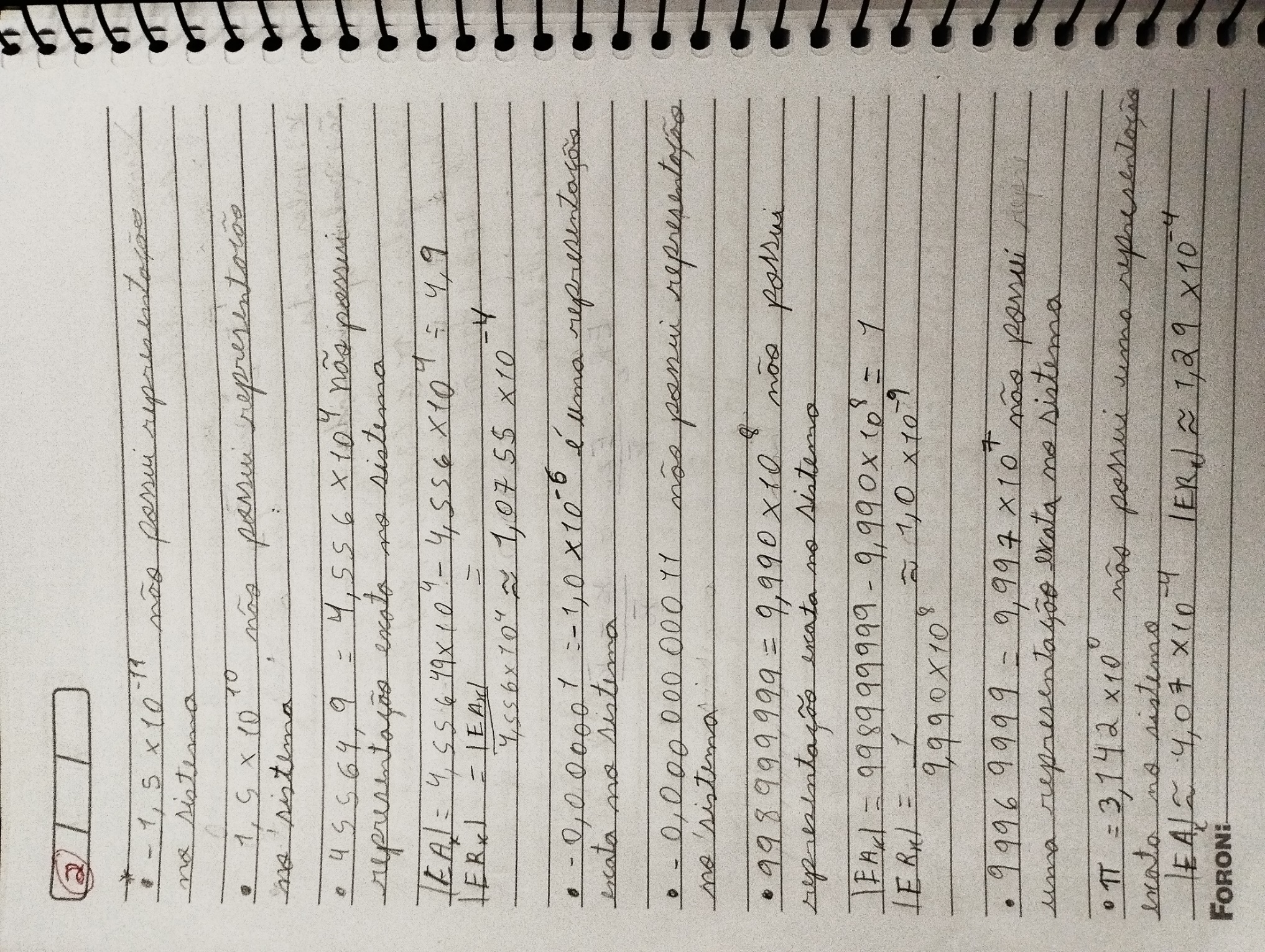
**Todos os arquivos estão disponíveis no seguinte repositório do GitHub:** [**https://github.com/AnubisNL/numerico/tree/main/Lista%20Fevereiro**](https://github.com/AnubisNL/numerico/tree/main/Lista%20Fevereiro)

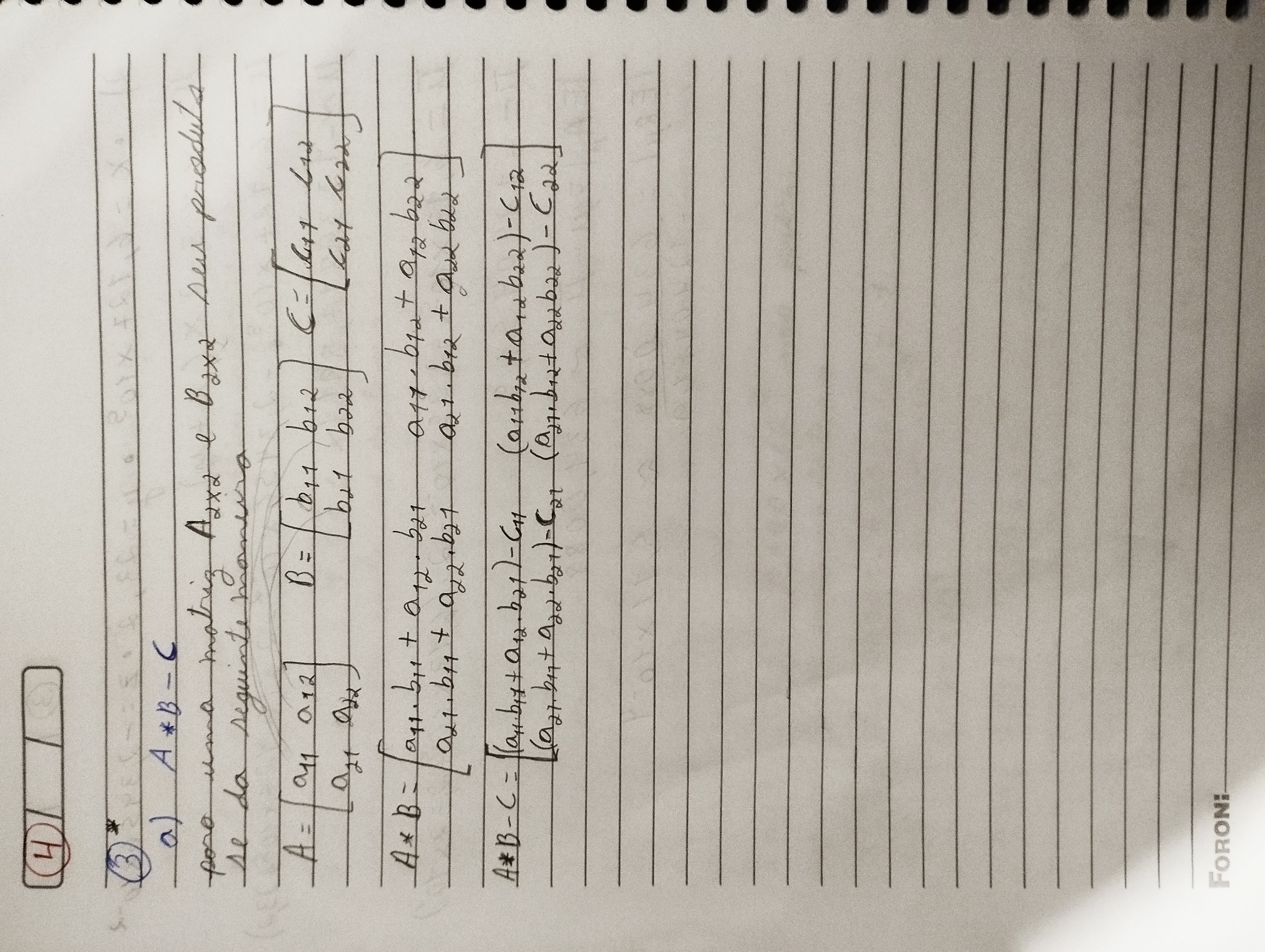
# Exercício 1

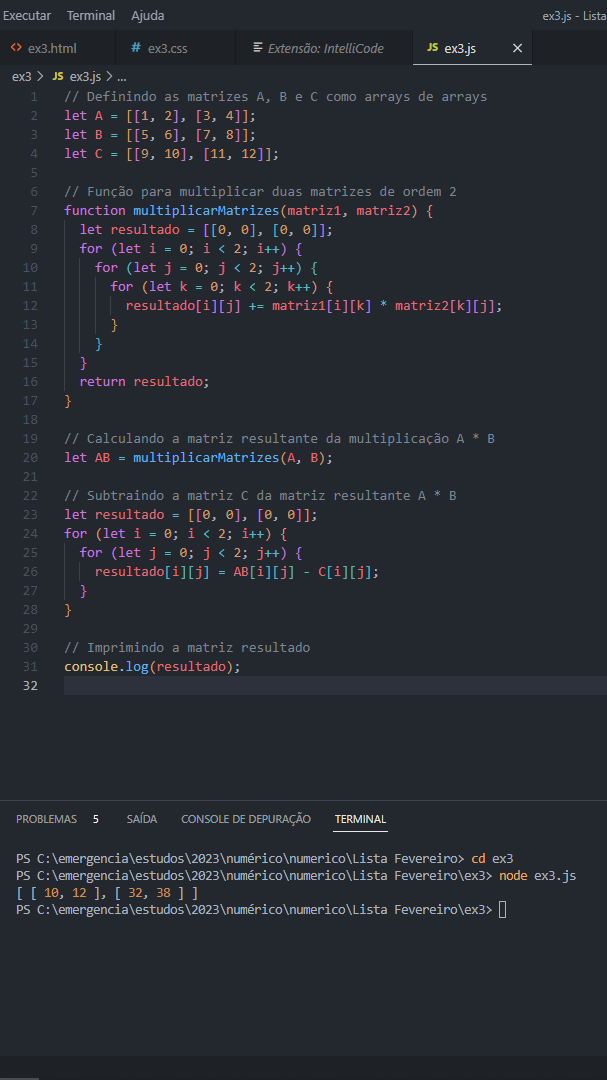


# Exercício 2)

Exercício 3)

## A)

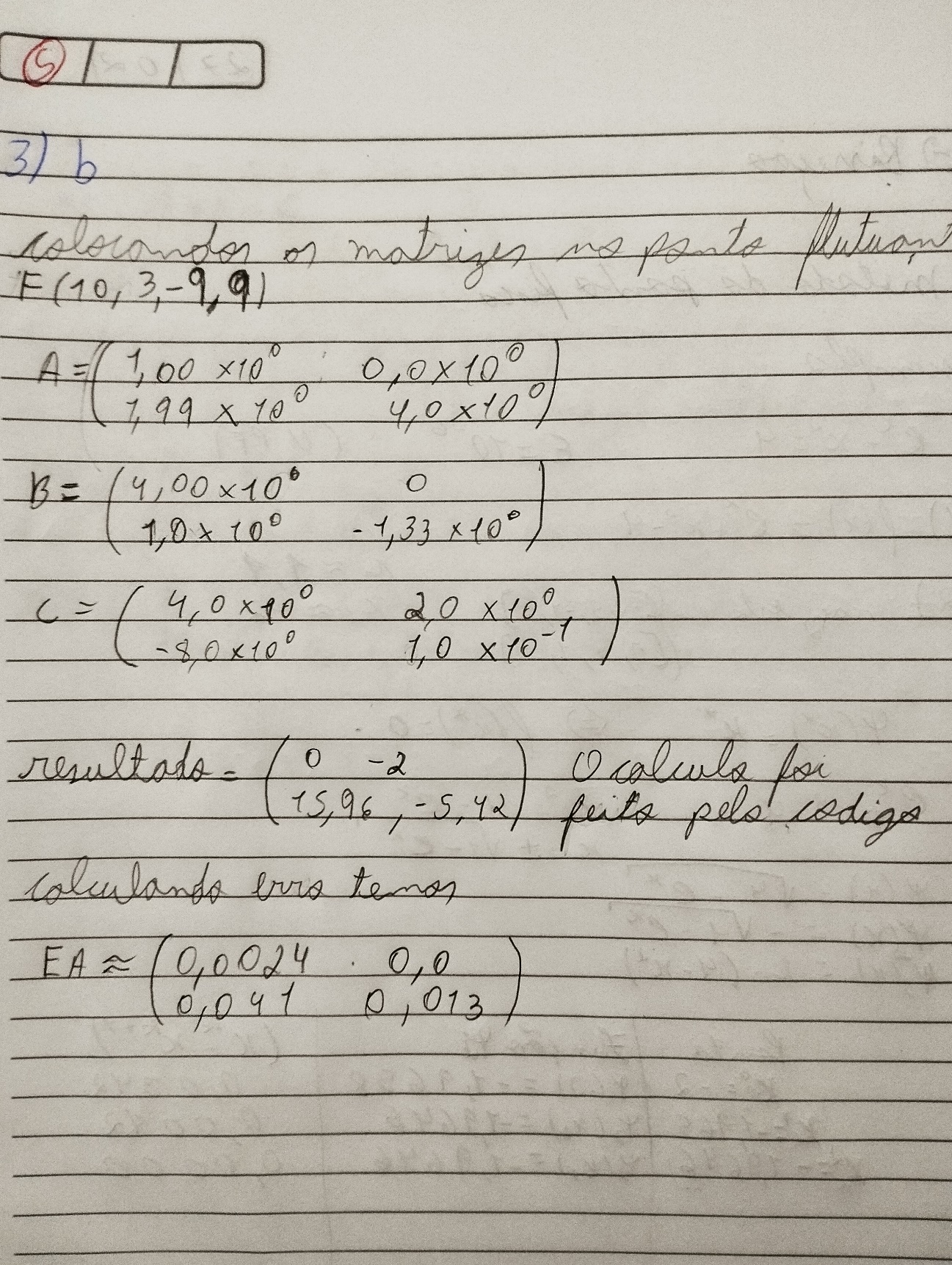


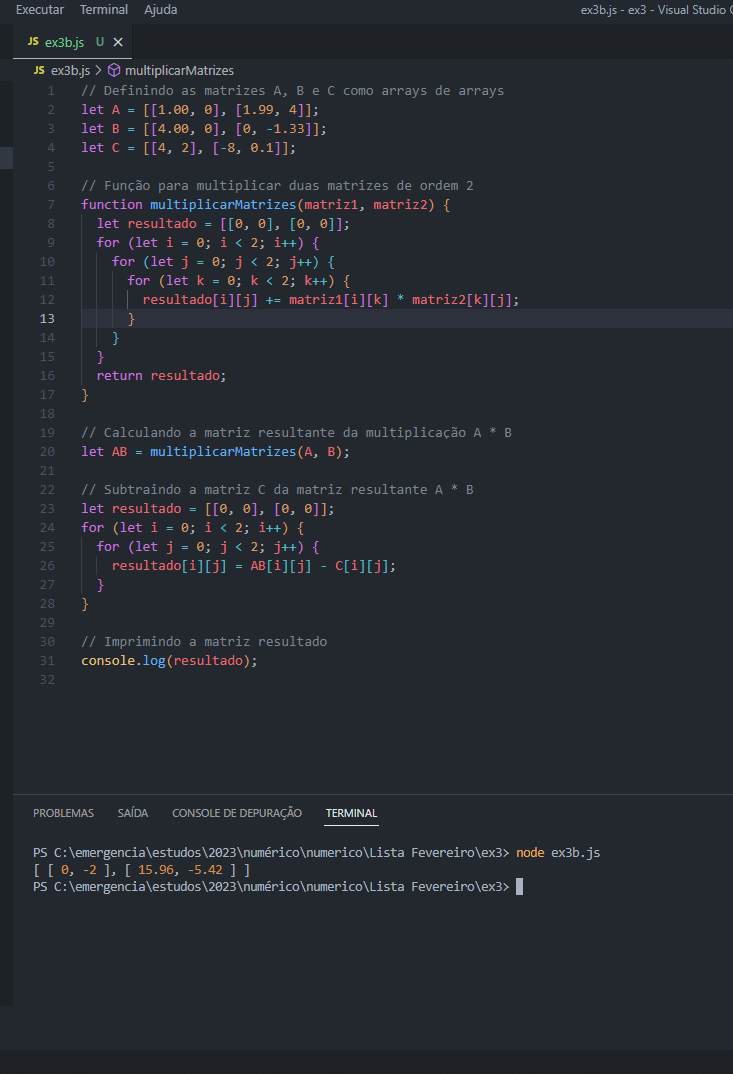


Exercício 3)

## B)

Em resumo, o algoritmo pode não ser estável para todas as matrizes de entrada e pode gerar erros significativos devido à limitação de precisão e intervalo da máquina F(10, 3, -9, 9).





Não podemos dizer que o algoritmo sempre será estável ao variar os coeficientes das matrizes. O problema é que alguns valores dos coeficientes podem levar a uma amplificação de erros de arredondamento que levarão a uma solução numérica imprecisa. Isso pode ocorrer, por exemplo, quando os coeficientes das matrizes são muito grandes ou muito pequenos em relação ao tamanho da matriz. Nesses casos, os erros de arredondamento que normalmente seriam insignificantes podem se tornar duradouros.

Um contra-exemplo numérico seria se tomarmos as matrizes A, B e C como:

A = [[1e15, 0], [0, 1]]

B = [[1, 0], [0, 1e-15]]

C = [[1, 0], [0, 1]]

Nesse caso, o algoritmo pode levar a um resultado impreciso devido à amplificação de erros de arredondamento. O produto A\*B é igual a:

A\*B = [[1e15, 0], [0, 1e-15]]

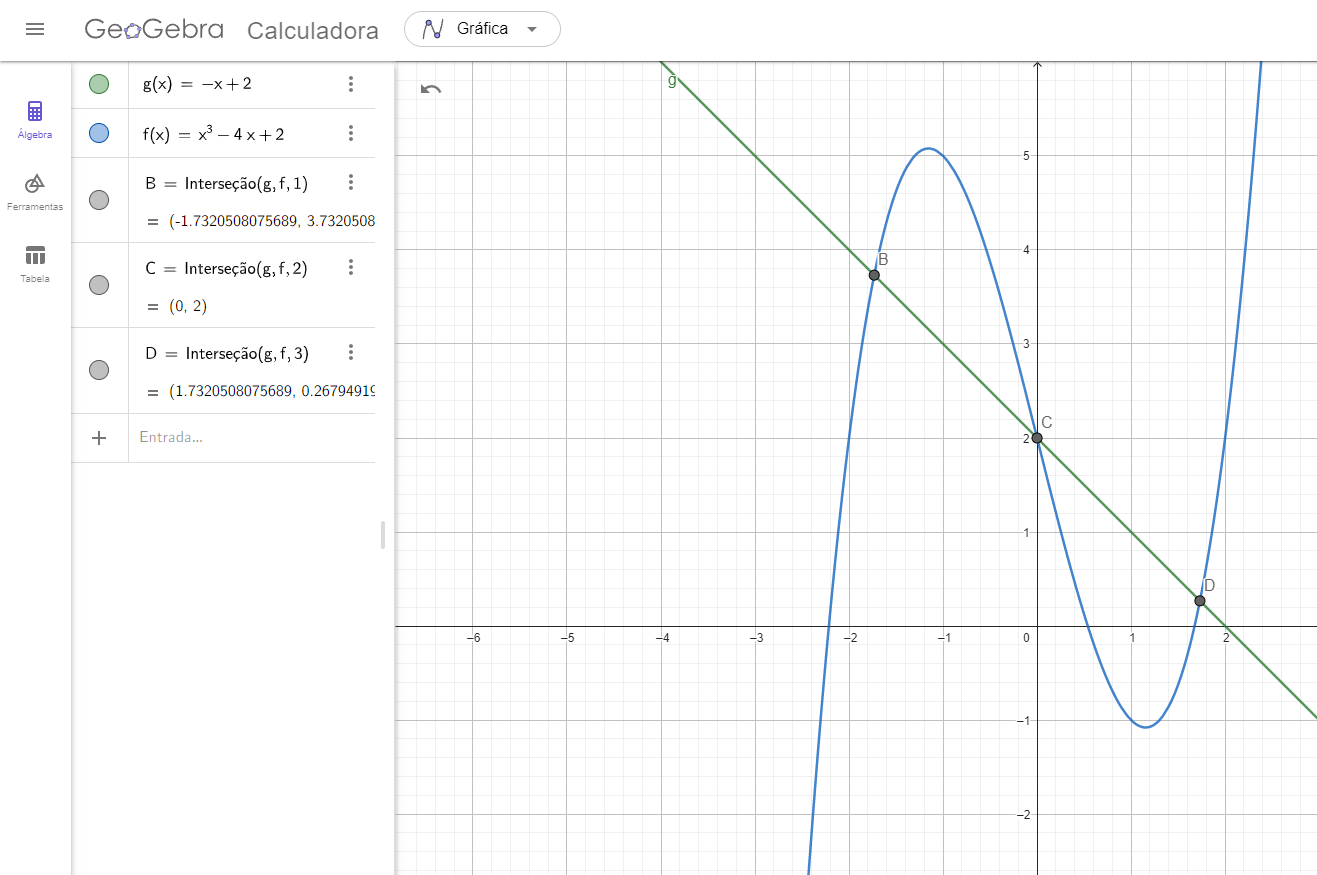
No entanto, a subtração A\*B - C é igual a:

A\*B - C = [[1e15 - 1, 0], [0, 1e-15 - 1]]

Observe que o elemento (1,1) da matriz resultante é muito menor do que o elemento correspondente na matriz A, mas ainda é grande em relação ao elemento correspondente na matriz C. Isso significa que o erro de arredondamento nesse elemento foi amplificado, gerado em uma solução imprecisa.

# Exercício 4)

## a)

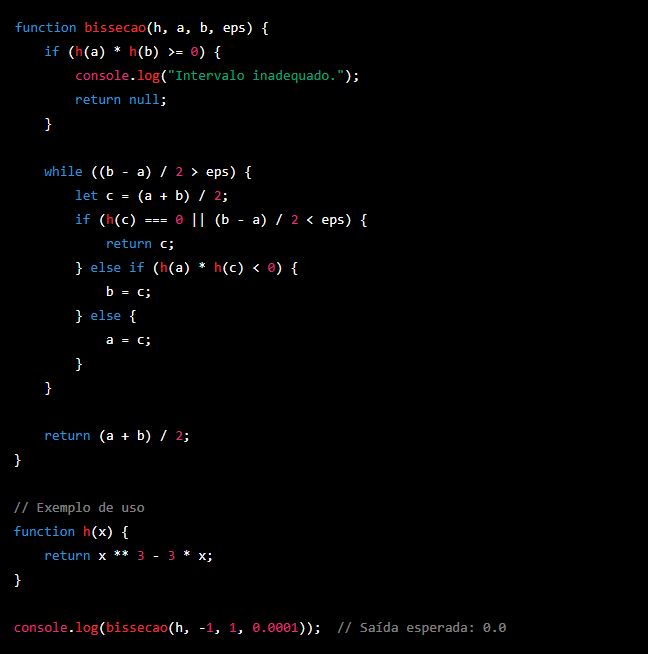


## b)

Algoritmo da bisseção:

1. Verifique se h(a) \* h(b) < 0. Se não, informe que o intervalo não é adequado e pare o algoritmo.
2. Enquanto (b - a) / 2 > ϵ: a. Calcule o ponto médio c = (a + b) / 2. b. Se h(c) = 0 ou (b - a) / 2 < ϵ, pare o algoritmo. c. Se h(a) \* h(c) < 0, atualize b = c. d. Senão, atualize a = c.
3. Retorne o valor de c como o zero da função h(x).

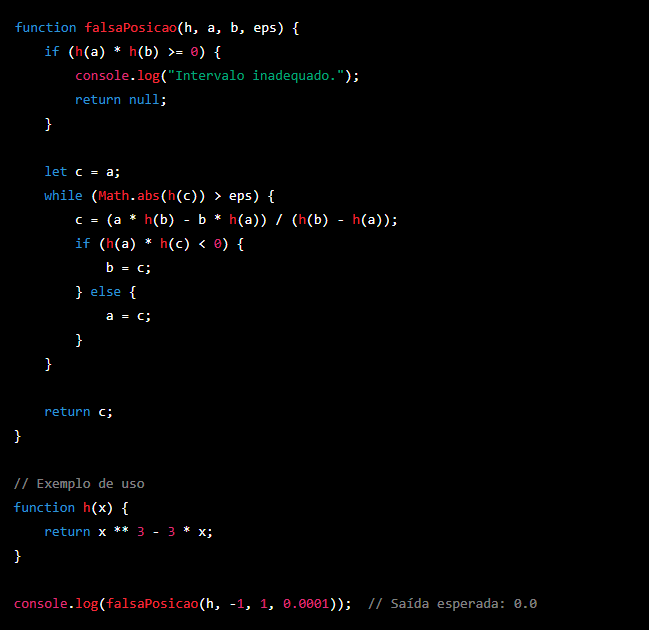
Código em Javascript:



Algoritmo da falsa posição:

1. Verifique se h(a) \* h(b) < 0. Se não, informe que o intervalo não é adequado e pare o algoritmo.
2. Enquanto |f(c)| > eps: a. Calcule o ponto c = (a \* h(b) - b \* h(a)) / (h(b) - h(a)). b. Se h(a) \* h(c) < 0, atualize b = c. c. Senão, atualize a = c.
3. Retorne o valor de c como o zero da função h(x).

Código em Javascript:



## c)

A fim de ler e executar o programa com as variáveis solidas pelo usuário, browser para rodar nosso script, com isso basta abrir o arquivo .HTML

* [−3,−1] -> o resultado usando o método de Bisseção é = -1.73199462890625

o resultado usando o método de Falsa Posição é = -1.7320365807780067

* [−1, 1] -> o resultado usando o método de Bisseção é = 0

o resultado usando o método de Falsa Posição é = 0

* [1, 3] -> o resultado usando o método de Bisseção é = 1.73199462890625

o resultado usando o método de Falsa Posição é = 1.7320365807780067

* [−1, 3]-> intervalo inadequado

## d)

Não os algoritmos não dão os mesmos resultados.

Em geral, a escolha entre os dois métodos depende da natureza da função que está sendo avaliada. Se a função é simples e bem comportada, a Bisseção pode ser a melhor opção. Se a função é mais complexa ou não é monotônica, a Falsa Posição pode ser mais eficiente.

# Exercício 5)

## a)

Para verificar analiticamente que a função h(x) = sin(x) - x + 1 possui apenas um zero real, podemos analisar suas propriedades matemáticas.

Começando pela sua derivada, temos:

h'(x) = cos(x) - 1

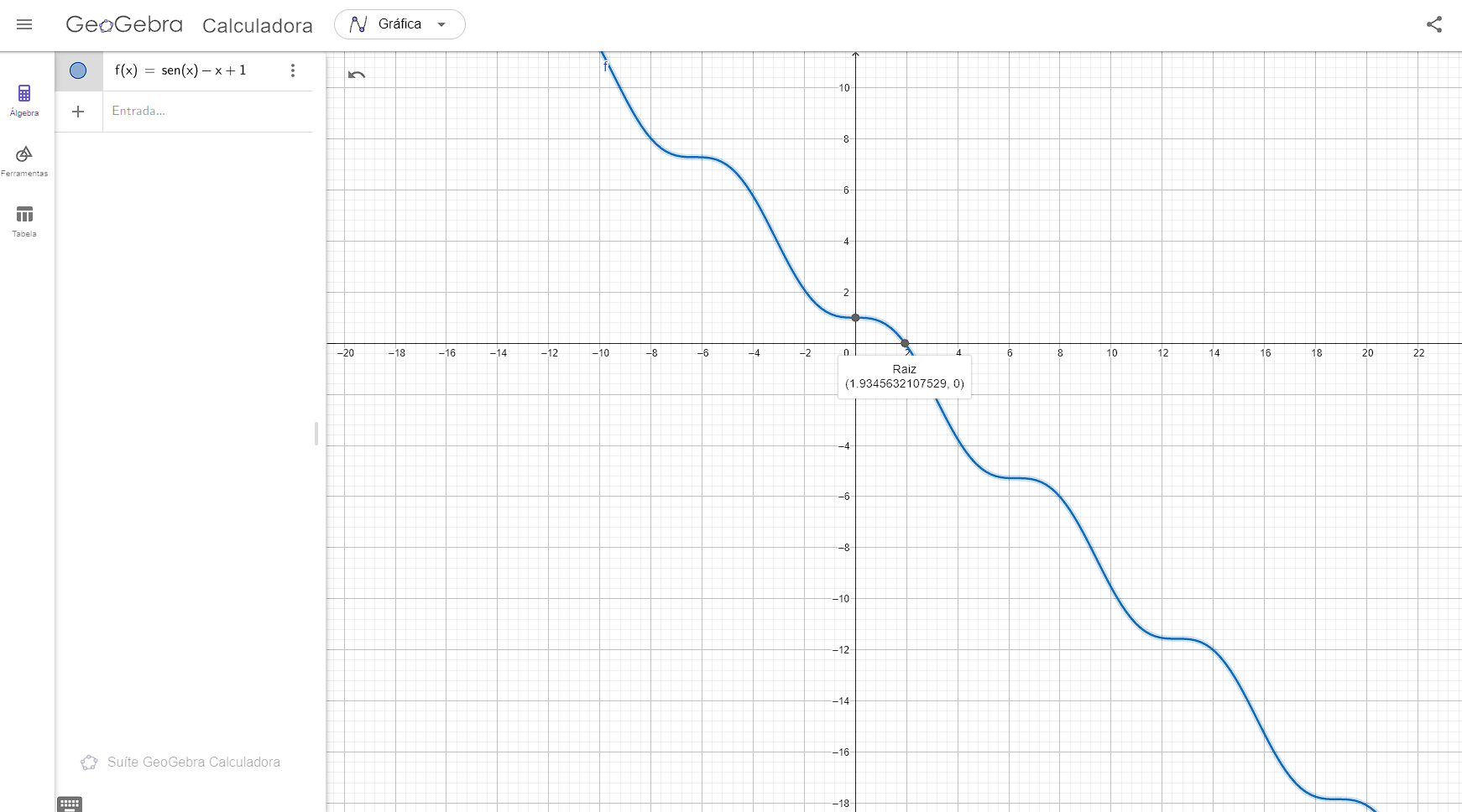
A derivada da função h(x) é h'(x) = cos(x) - 1. Notamos que esta derivada é sempre negativa para valores de x menores que π/2 e sempre positiva para valores maiores que π/2. Além disso, a derivada é zero apenas quando x = π/2.

Isso significa que a função h(x) é decrescente no intervalo (-∞,π/2] e crescente no intervalo [π/2,+∞). Como h(x) é uma função contínua, ela não pode mudar de sinal sem passar por zero, o que implica que há no máximo um zero em cada um desses intervalos.

Além disso, podemos observar que h(x) é estritamente crescente em [π/2,+∞), o que implica que há exatamente um zero em [π/2,+∞), já que h(π/2) > 0.

Já em (-∞,π/2], podemos notar que h(0) = 1 e h(π/2) > 0, o que implica que há pelo menos um zero neste intervalo. No entanto, como h(x) é uma função contínua, ela não pode ter mais do que um zero nesse intervalo, já que h(x) muda de sinal apenas uma vez.

Portanto, podemos concluir que a função h(x) = sin(x) - x + 1 possui apenas um zero real ξ, o que pode ser verificado graficamente observando que o gráfico da função intersecta o eixo x apenas uma vez.



## b)

