

Trabalho Aplicado 1

Erickson Giesel Müller

23 de Setembro de 2024

1 Código

```
import sympy

def somaRiemann(funcao, a, b, n):
    x = sympy.Symbol('x')
    f = sympy.sympify(funcao)

    delta_x = (b - a) / n
    area = 0
    for i in range(n):
        x_i = a + i * delta_x + delta_x / 2
        altura = f.subs(x, x_i)
        area += altura * delta_x

    return area

funcao = input("escreva_a_funcao_f(x):_")
a = int(input("Insira_o_valor_de_a:_"))
b = int(input("Insira_o_valor_de_b:_"))
n = int(input("insira_o_valor_de_n:_"))

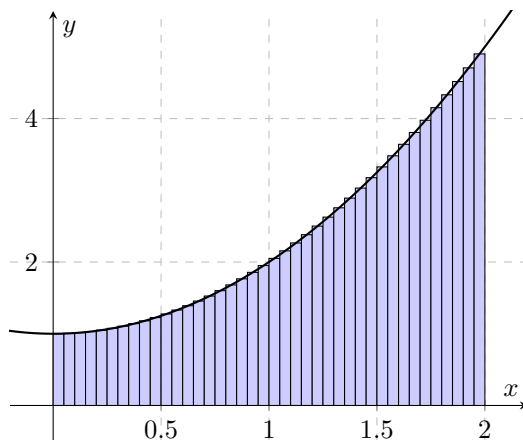
aprox_area = somaRiemann(funcao, a, b, n)
print("A_area_aproximada_usando_a_Soma_de_Riemann
e: {:.6f}_u.a".format(aprox_area))
```

Para calcular a aproximação da área abaixo da curva, o código em python usou a biblioteca *sympy*, que fornece diversas ferramentas de álgebra computacional. Para instalar a biblioteca, caso o usuário tenha o python instalado em seu computador, basta rodar no terminal o comando:

```
pip install sympy
```

2 Soma de Riemann

O programa calcula a aproximação da área fazendo a soma das áreas dos n retângulos abaixo da curva. Os retângulos tem base $\frac{b-a}{n}$ e altura $f(x)$ no ponto médio. Semelhante à figura abaixo:



3 Como operar o programa

Para usar o programa no prompt de comando, o usuário deverá escrever:

```
python t1-20230001178.py
```

Em seguida serão solicitadas os dados de entrada:

$y; a; b; n$

A função é calculada no formato de string, portanto o usuário deverá escrevê-la no seguinte formato:

```
a*x**n+b*x**(n-1)+c*x**(n-2)...+z
```

Para representar o número e , deve-se digitar a letra maiúscula E , pois essa é a sintaxe da biblioteca *sympy*.

4 Resolução dos problemas:

Considerando a função $f(x) = e^{-x} + 1$, calcular as aproximações das áreas no intervalo $[-1, 1]$ com n subintervalos:

4.1 $n=4$

```
~/Documents/notes-UFFS/Cálculo II/Trabalho Aplicado
[Erickson@Host Trabalho Aplicado]$ python t1-20230001178.py
escreva a função f(x): E**-x+1
Insira o valor de a: -1
Insira o valor de b: 1
insira o valor de n: 4
A área aproximada usando a Soma de Riemann é: 4.326096 unidades de área
[Erickson@Host Trabalho Aplicado]$ F
```

4.2 $n=10$

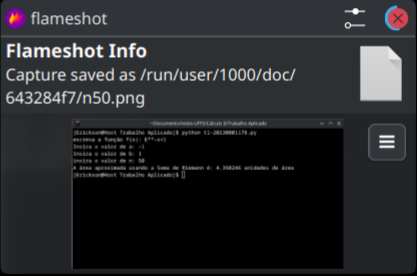
```
~/Documents/notes-UFFS/Cálculo II/Trabalho Aplicado
[Erickson@Host Trabalho Aplicado]$ python t1-20230001178.py
escreva a função f(x): E**-x+1
Insira o valor de a: -1
Insira o valor de b: 1
insira o valor de n: 10
A área aproximada usando a Soma de Riemann é: 4.346490 unidades de área
[Erickson@Host Trabalho Aplicado]$
```

4.3 $n=50$

```
~/Documents/notes-UFFS/Cálculo II/Trabalho Aplicado
[Erickson@Host Trabalho Aplicado]$ python t1-20230001178.py
escreva a função f(x): E**-x+1
Insira o valor de a: -1
Insira o valor de b: 1
insira o valor de n: 50
A área aproximada usando a Soma de Riemann é: 4.350246 unidades de área
[Erickson@Host Trabalho Aplicado]$
```

4.4 $n=1000$

```
~/Documents/notes-UFFS/Cálculo II/Trabalho Aplicado
[Erickson@Host Trabalho Aplicado]$ python t1-20230001178.py
escreva a função f(x): E**-x+1
Insira o valor de a: -1
Insira o valor de b: 1
insira o valor de n: 1000
A área aproximada usando a Soma de Riemann é:
[Erickson@Host Trabalho Aplicado]$
```



5 Considerações Finais

Analisando os resultados, percebemos que quanto maior o número de subintervalos, mais preciso será o resultado da aproximação em relação à integral. Isso acontece porque a área dos retângulos que está sendo calculada a uma distância h_i da curva é reduzida conforme a largura da base dos retângulos diminui.