

Eduardo Brunaldi dos Santos — 8642515,  
Jorge Ashkar Ferreira Simondi — 8517081,  
Victor Luiz da Silva Mariano Pereira — 8602444

**Trabalho 2**  
**Método de Integração Numérica**  
**Simpson 1/3 Composta**

Brasil

2018



Eduardo Brunaldi dos Santos — 8642515,  
Jorge Ashkar Ferreira Simondi — 8517081,  
Victor Luiz da Silva Mariano Pereira — 8602444

**Trabalho 2**  
**Método de Integração Numérica**  
**Simpson 1/3 Composta**

Universidade de São Paulo – USP  
Instituto de Ciências Matemáticas e de Computação – ICMC  
Cálculo Numérico – SME0104

Professor Murilo Francisco Tomé

Brasil  
2018

# Conteúdo

	<b>Introdução</b> . . . . .	<b>3</b>
<b>1</b>	<b>INTEGRAÇÃO NUMÉRICA SIMPSON 1/3 COMPOSTA</b> . . . . .	<b>5</b>
<b>2</b>	<b>RESULTADOS</b> . . . . .	<b>7</b>
	<b>Conclusão</b> . . . . .	<b>9</b>
	 <b>APÊNDICES</b>	 <b>11</b>
	<b>APÊNDICE A – CÓDIGOS FONTE</b> . . . . .	<b>13</b>
<b>A.1</b>	<b>Programa principal (main.c)</b> . . . . .	<b>13</b>
<b>A.2</b>	<b>Biblioteca auxiliar</b> . . . . .	<b>13</b>
A.2.1	<i>Header da biblioteca (simpson_composta.h)</i> . . . . .	13
A.2.2	<i>Implementação da biblioteca (simpson_composta.c)</i> . . . . .	15

# Introdução



# 1 Integração numérica Simpson $1/3$ composta





## 2 Resultados



## Conclusão



## Apêndices



# APÊNDICE A – Códigos Fonte

## A.1 Programa principal (main.c)

---

```

1  /**
2   *   Trabalho 2 - Método de Integração Numérica Simpson 1/3 Composta
3   *
4   *   Cálculo Numérico   SME-0104
5   *   Prof.: Murilo Francisco Tomé
6   *
7   *   Eduardo Brunaldi dos Santos           8642515
8   *   Jorge Ashkar Ferreira Simondi         8517081
9   *   Victor Luiz da Silva Mariano Pereira  8602444
10  */
11
12  #include <stdio.h>
13  #include <stdlib.h>
14  #include <simpson_composta.h>
15
16  int main (int argc, char *argv[]){
17      long double x0;
18      long double xN;
19      long double n;
20
21      scanf("%Lf", &x0);
22      scanf("%Lf", &xN);
23      scanf("%Lf", &n);
24
25      printf("%.16Lf\n", simpson_composta(x0, xN, n, funcao_f));
26      return 0;
27  }
```

---

## A.2 Biblioteca auxiliar

### A.2.1 Header da biblioteca (simpson\_composta.h)

---

```

1  /**
2   *   Trabalho 2 - Método de Integração Numérica Simpson 1/3 Composta
3   *
4   *   Cálculo Numérico   SME-0104
5   *   Prof.: Murilo Francisco Tomé
```

---

```

6  *
7  *      Eduardo Brunaldi dos Santos      8642515
8  *      Jorge Ashkar Ferreira Simondi    8517081
9  *      Victor Luiz da Silva Mariano Pereira 8602444
10 */
11
12 #ifndef SIMPSON_COMPOSTA_H
13 #define SIMPSON_COMPOSTA_H
14
15 /**
16  * Definindo um ponteiro para uma função para poder chamar a função desejada
17  * por parâmetro
18  */
19 typedef long double (*Funcao)(long double);
20
21 /**
22  * Função que calcula o valor de  $f(x)$ , sendo  $f(x)$  dada por:
23  *
24  *       $f(x) = (1/\sqrt{2}) * e^{((-x^2)/2)}$ 
25  *
26  * @param x ponto x onde será calculada  $f(x)$ 
27  * @return valor da função  $f(x)$  calculada no ponto x
28  */
29 long double funcao_f (long double x);
30
31 /**
32  * Função para calcular a integração de uma certa função f usando o método de
33  * Simpson 1/3 Composta, dada por:
34  *
35  *       $I_N_S = (h/3) * (f(x_0) + f(x_N) + 4 * SUM(f(x_{\text{impares}})) + 2 * SUM(f(x_{\text{pares}})))$ 
36  *
37  * Utilizando um intervalo, a quantidade de divisões do intervalo e a função a
38  * ser integrada.
39  *
40  * @param x0 valor inicial do intervalo
41  * @param xN valor final do intervalo
42  * @param n quantas vezes será dividido o intervalo
43  * @param f função a ser integrada
44  * @return valor aproximado da integral da função f
45  */
46 long double simpson_composta(long double x0, long double xN, int n, Funcao f);
47
48 #endif

```

---



## A.2.2 Implementação da biblioteca (simpson\_composta.c)

---

```

1  /**
2   *   Trabalho 2 - Método de Integração Numérica Simpson 1/3 Composta
3   *
4   *   Cálculo Numérico   SME-0104
5   *   Prof.: Murilo Francisco Tomé
6   *
7   *   Eduardo Brunaldi dos Santos           8642515
8   *   Jorge Ashkar Ferreira Simondi        8517081
9   *   Victor Luiz da Silva Mariano Pereira  8602444
10  */
11
12  #include <stdio.h>
13  #include <stdlib.h>
14  #include <math.h>
15  #include <simpson_composta.h>
16
17  /**
18   * Função que calcula o valor de  $f(x)$ , sendo  $f(x)$  dada por:
19   *
20   *    $f(x) = (1/\sqrt{2}) * e^{(-x^2)/2}$ 
21   *
22   * @param x ponto x onde será calculada  $f(x)$ 
23   * @return valor da função  $f(x)$  calculada no ponto x
24   */
25  long double funcao_f (long double x){
26      return (1.0/(sqrt(2*M_PI)*1.0))*exp((- pow(x, 2.0))/2.0);
27  }
28
29  /**
30   * Função para calcular a integração de uma certa função f usando o método de
31   * Simpson 1/3 Composta, dada por:
32   *
33   *  $I_N^S = (h/3) * (f(x_0) + f(x_N) + 4 * SUM(f(x_{\text{impares}})) + 2 * SUM(f(x_{\text{pares}})))$ 
34   *
35   * Utilizando um intervalo, a quantidade de divisões do intervalo e a função a
36   * ser integrada.
37   *
38   * @param x0 valor inicial do intervalo
39   * @param xN valor final do intervalo
40   * @param n quantas vezes será dividido o intervalo
41   * @param f função a ser integrada
42   * @return valor aproximado da integral da função f
43   */
44  long double simpson_composta(long double x0, long double xN, int n, Funcao f){
45      long double x;

```

```
46     long double h;
47     long double resposta;
48     long double pares;
49     long double impares;
50
51     int i;
52
53     if (n % 2 != 0)
54         n += 1;
55     h = (xN - x0)/n;
56
57     pares = 0;
58     impares = 0;
59     x = x0 + h;
60
61     resposta = f(x0) + f(xN);
62
63     for (i = 1; i < n; i++){
64         if (i % 2 == 0)
65             pares += f(x);
66         else
67             impares += f(x);
68
69         x += h;
70     }
71
72     resposta += 4 * impares + 2 * pares;
73     resposta *= h/3;
74
75     return resposta;
76 }
```

---