

プログラミング演習I 課題ドキュメント

b162392

構築: Doxygen 1.8.6

2016 年 12 月 15 日 (木) 15 時 21 分 05 秒

Contents

1	ファイル詳解	1
1.1	main.c ファイル	1
1.1.1	詳解	1
1.1.2	関数詳解	1
1.2	main.c	3

1 ファイル詳解

1.1 main.c ファイル

矩形近似による数値積分

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

関数

- double [linear](#) (double x)
1 次関数
- double [quad](#) (double x)
2 次関数
- int [main](#) (int argc, char *argv[])
矩形近似による数値積分

1.1.1 詳解

矩形近似による数値積分

日付

2016/12/15

著者

佐伯雄飛
B162392

[main.c](#) に定義があります。

1.1.2 関数詳解

1.1.2.1 double linear (double x)

1 次関数

[main.c](#) の 行目に定義があります。

```
00016      {  
00017  double a = 2.0, b = 0.5;  
00018  return a * x + b;  
00019 }
```

1.1.2.2 int main (int argc, char * argv[])

矩形近似による数値積分

次の近似式による数値積分を行う． $\int_a^b f(x)dx \approx \sum_{i=1}^n f(x_i) \frac{b-a}{n}$

コマンドラインオプション：

- 1 目：被積分関数 f(x) (linear, quad, sin, con のいずれか)
- 2 目：a
- 3 目：b , ただし $a < b$
- 4 目：n , ただし $0 < n$

出力：

- 積分結果を小数点第 5 位まで表示 (%.5f)
- 次の場合は，何も表示せずに終了する (return 0 で) ． 被積分関数のオプションに上記 4 つの関数以外を指定した場合． a が b 以上の場合． n が負の場合．

実行例：

```
./main linear 0 10 1000
```

出力例：

```
105.10500
```

実行例：

```
./main sin 0 10 100000
```

出力例：

```
1.83904
```

実行例：

```
./main log 0 10 100
```

出力例：

日付

2016/12/15

著者

佐伯雄飛, B162392

main.c の行目に定義があります。

```
00073                                     {
00074     if (argc < 5) {
00075         return 0;
00076     }
00077
00078     double (*f)(double); // function pointer
00079
00080     if (strcmp("linear", argv[1], 10) == 0) {
00081         f = linear;
00082     } else if (strcmp("quad", argv[1], 10) == 0) {
00083         f = quad;
00084     } else if (strcmp("sin", argv[1], 10) == 0) {
00085         f = sin;
00086     } else if (strcmp("cos", argv[1], 10) == 0) {
00087         f = cos;
00088     } else {
00089         return 0;
00090     }
00091
00092     double a = atof(argv[2]);
00093     double b = atof(argv[3]);
00094     if (a >= b) {
00095         return 0;
00096     }
00097     int n = atoi(argv[4]);
00098     if (n <= 0) {
00099         return 0;
00100     }
00101
00102     double h = (b - a) / n;
00103     double integral = 0.0;
00104
00105     for (double x = a; x <= b; x += h) {
00106         integral += f(x) * h;
00107     }
00108
00109     printf("%.5f\n", integral);
00110
00111     return 0;
00112 }
```

1.1.2.3 double quad (double x)

2 次関数

main.c の行目に定義があります。

```
00023                                     {
00024     double a = 2.0, b = 0.5, c = 1.0;
00025     return a * x * x + b * x + c;
00026 }
```

1.2 main.c

```
00001 /** @file main.c
00002  * @brief 矩形近似による数値積分
00003  * @date 2016/12/15
00004  * @author 佐伯雄飛
00005  * @author B162392
00006  */
00007
00008 #include <stdio.h>
00009 #include <stdlib.h> // atoi, atof
00010 #include <math.h> // sin, cos
00011 #include <string.h> // strcmp
00012
00013 /** @fn double linear(double x)
00014  * @brief 1 次関数
00015  */
00016 double linear(double x) {
00017     double a = 2.0, b = 0.5;
00018     return a * x + b;
00019 }
```

```

00019 }
00020 /** @fn double quad(double x)
00021 * @brief 2 次関数
00022 */
00023 double quad(double x) {
00024     double a = 2.0, b = 0.5, c = 1.0;
00025     return a * x * x + b * x + c;
00026 }
00027
00028 /** @fn int main(int argc, char* argv[])
00029 * @brief 矩形近似による数値積分
00030 *
00031 * 次の近似式による数値積分を行う .
00032 *  $\int_a^b f(x) dx \approx \sum_{i=1}^n f(x_i) \frac{b-a}{n}$ 
00033 *
00034 * コマンドラインオプション :
00035 * - 1 目 : 被積分関数 f(x) (linear, quad, sin, cos のいずれか)
00036 * - 2 目 : a
00037 * - 3 目 : b, ただし a < b
00038 * - 4 目 : n, ただし 0 < n
00039 *
00040 * 出力 :
00041 * - 積分結果を小数点第 5 位まで表示 (%.5f)
00042 * - 次の場合は, 何も表示せずに終了する (return 0 で) .
00043 *   被積分関数のオプションに上記 4 つの関数以外を指定した場合 .
00044 *   a が b 以上の場合 .
00045 *   n が負の場合 .
00046 *
00047 * 実行例 :
00048 \verbatim
00049 ./main linear 0 10 1000
00050 \endverbatim
00051 * 出力例 :
00052 \verbatim
00053 105.10500
00054 \endverbatim
00055 * 実行例 :
00056 \verbatim
00057 ./main sin 0 10 100000
00058 \endverbatim
00059 * 出力例 :
00060 \verbatim
00061 1.83904
00062 \endverbatim
00063 * 実行例 :
00064 \verbatim
00065 ./main log 0 10 100
00066 \endverbatim
00067 * 出力例 :
00068 \verbatim
00069 \endverbatim
00070 * @date 2016/12/15
00071 * @author 佐伯雄飛, B162392
00072 */
00073 int main(int argc, char* argv[]) {
00074     if (argc < 5) {
00075         return 0;
00076     }
00077
00078     double (*f)(double); // function pointer
00079
00080     if (strcmp("linear", argv[1], 10) == 0) {
00081         f = linear;
00082     } else if (strcmp("quad", argv[1], 10) == 0) {
00083         f = quad;
00084     } else if (strcmp("sin", argv[1], 10) == 0) {
00085         f = sin;
00086     } else if (strcmp("cos", argv[1], 10) == 0) {
00087         f = cos;
00088     } else {
00089         return 0;
00090     }
00091
00092     double a = atof(argv[2]);
00093     double b = atof(argv[3]);
00094     if (a >= b) {
00095         return 0;
00096     }
00097     int n = atoi(argv[4]);
00098     if (n <= 0) {
00099         return 0;
00100     }
00101
00102     double h = (b - a) / n;
00103     double integral = 0.0;
00104
00105     for (double x = a; x <= b; x += h) {

```

```
00106     integral += f(x) * h;
00107 }
00108
00109 printf("%.5f\n", integral);
00110
00111 return 0;
00112 }
```