



Итоговая работа по курсу «Программирование на языке С (продвинутый уровень)»

Мельков Евгений Валерьевич

Постановка задания

Создание консольного приложения — Численное интегрирование

С заданной точностью ε вычислить площадь плоской фигуры, ограниченной тремя кривыми, уравнения которых $y = f_1(x)$, $y = f_2(x)$ и $y = f_3(x)$:

- $f_1 = 0.6x + 3$
- $f_2 = (x - 2)^3 - 1$
- $f_3 = 3 / x$

Требования к программе

1. Основная программа должна поддерживать опции командной строки, при задании которых:
 - a. Печатаются абсциссы точек пересечения кривых
 - b. Печатается число итераций, потребовавшихся на приближённое решение уравнений при поиске точек пересечения
2. Программа должна поддерживать ключ командной строки `-h`, выводящий на печать все допустимые ключи командной строки
3. Вычисление с точностью ε_1 корня x уравнения $f(x) = g(x)$ на отрезке $[a, b]$ должно быть реализовано в отдельной Си-функции `root(f, g, a, b, eps1)`. Если используется метод касательных или комбинированный метод, то у `root` должно быть ещё два параметра функционального типа, позволяющие вызывать производные функций f и g
4. Вычисление с точностью ε_2 величины определенного интеграла от функции $f(x)$ на отрезке $[a, b]$ должно быть реализовано в отдельной Си-функции `integral(f, a, b, eps2)`
5. Си-функции `root` и `integral` должны быть предварительно протестированы. Основная программа должна предоставлять возможности тестирования, активируемые опцией командной строки

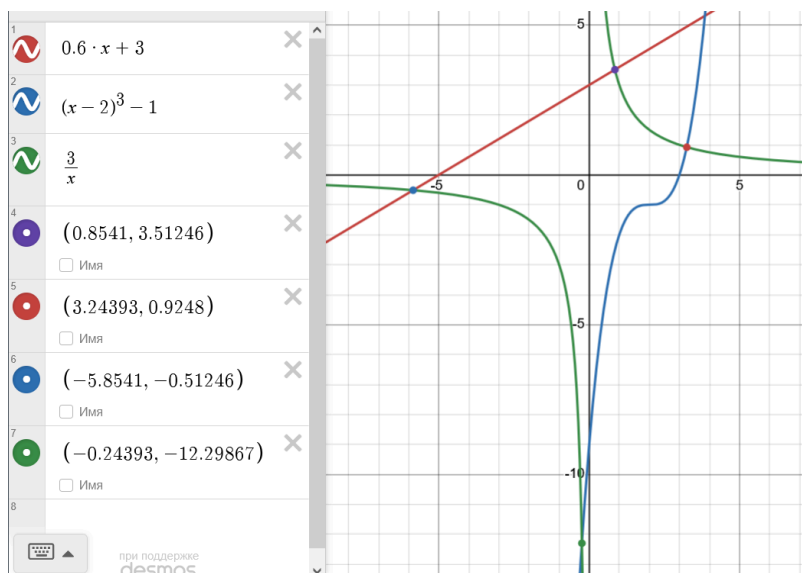
Требования к сборке приложения

- Приложение должно собираться при помощи утилиты `make`.

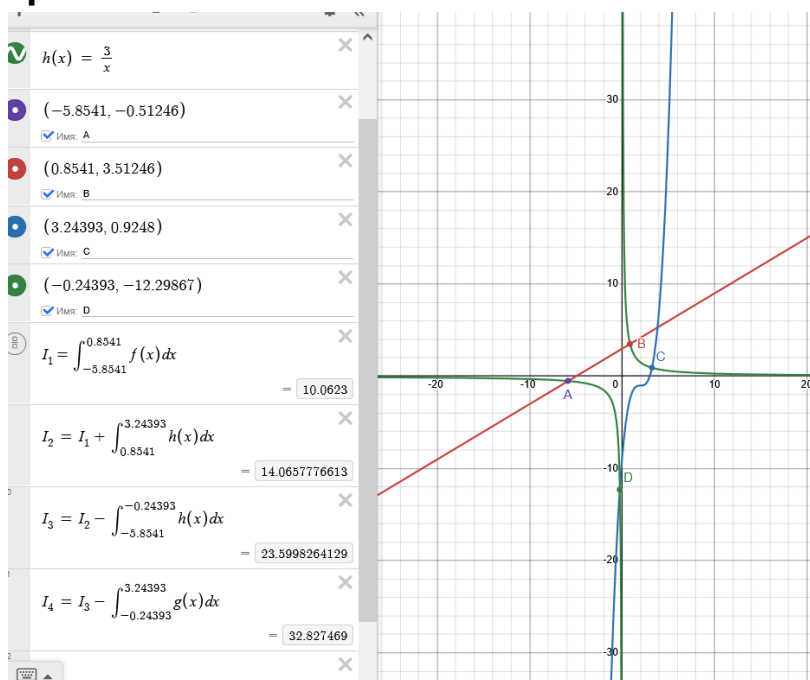
Описание программы

Подготовка

Графики функций и точки пересечения найденные с помощью сторонних программ.



Нахождение определенных интегралов для тестирования приложения




Исходный код программы




Ссылка на репозиторий


https://github.com/evm2772/home_c/tree/main/AC_CW


Описание файлов программы


 **Инструкция по оформлению:** В этом блоке напишите из каких файлов состоит ваш проект, а также кратко назначение каждого файла.


Программа состоит из следующих файлов:


 **integral.c** - основной файл программы с точкой входа, функцией main. Содержит логику запуска функций и обработку ключей, поступающих от командной строки;


 **itrapez.h** – файл прототипов функций численного интегрирования методом трапеций;

 **itrapez.c** – файл описания функций численного интегрирования методом трапеций;

 **roots.h** – файл прототипов функций для численного нахождения корней уравнения на отрезке;

 **roots.c** – файл описания функций для численного нахождения корней уравнения на отрезке;

 **Makefile** – файл инструкция для утилиты сборки mingw32-make(win) или make(linux);

 **integral** – собранный исполняемый файл программы (Linux)

Описание работы программы

Программа является консольным приложением и рекомендуется запускать её из командной строки.

При запуске без указания ключей, программа вычисляет площадь фигуры с заданной по умолчанию точностью.

При запуске при неправильном ключе , будет выведен help для получения инструкций.

Для получения подробной информации используйте команду

```
integral -h
```

Примеры написания командной строки для запуска:

```
integral
```

```
integral -h
```

```
integral -e 0.001
```

```
integral -e 0.0001 -r
```

Демонстрация работы программы

Образцы снимков (Linux):

Рисунок 1. Запуск программы с опцией «-h»

```
happyjoy@deepspace:~/MIPT/home_c/AC_CW$ ./integral -h
Numerical integration application.
-----
Numerical integration is used to calculate a numerical approximation for the value S ,
the area under the curves defined by:
f1(x) = 0.6 * x + 3
f2(x) = pow ((x - 2.), 3) - 1
f3(x) = 3/x
-----
Usage: integral [-h] [-e <epsilon value>] [r]
      -h      Show this help.
      -t      Test.
      -e      Accuracy. Epsilon is a float value. Default value is 0.00001
      -r      Roots. Points of intersection of the functions plots.
Examples:
      integral -h
      integral -t
      integral -r -e 0.01
      integral -e 0.001
happyjoy@deepspace:~/MIPT/home_c/AC_CW$
```

Рисунок 2. Запуск программы без ключей:

```
happyjoy@deepspace:~/MIPT/home_c/AC_CW$ ./integral
epsilon = 0.000010
----- Graphs Intersections -----
Steps number: 16
[1]Intersection f2 and f3 [-0.500000; -0.000010] ~ (-0.243928, -12.298660)

Steps number: 19
[2]Intersection f2 and f3 [2.000000; 6.000000] ~ (3.243931, 0.924814)

Steps number: 20
[1]Intersection f1 and f3 [-10.000000; -0.000010] ~ (-5.854101, -0.512460)

Steps number: 20
[2]Intersection f1 and f3 [0.000010; 10.000000] ~ (0.854106, 3.512463)

----- Figure Area -----
Area value = 32.827475
happyjoy@deepspace:~/MIPT/home_c/AC_CW$
```

Рисунок 3. Запуск программы вычисления интеграла с заданной точностью:

```
happyjoy@deepspace:~/MIPT/home_c/AC_CW$ ./integral -e 0.001

epsilon = 0.001000
----- Graphs Intersections -----
Steps number: 9
[1]Intersection f2 and f3 [-0.500000; -0.000010] ~ (-0.243657, -12.294569)

Steps number: 12
[2]Intersection f2 and f3 [2.000000; 6.000000] ~ (3.243652, 0.923521)

Steps number: 14
[1]Intersection f1 and f3 [-10.000000; -0.000010] ~ (-5.854191, -0.512515)

Steps number: 14
[2]Intersection f1 and f3 [0.000010; 10.000000] ~ (0.854196, 3.512518)

----- Figure Area -----
Area value = 32.827870
```

Рисунок 4. Запуск программы с нахождением точек пересечения

```
happyjoy@deepspace:~/MIPT/home_c/AC_CW$ ./integral -r

epsilon = 0.000010
----- Graphs Intersections -----
Steps number: 16
[1]Intersection f2 and f3 [-0.500000; -0.000010] ~ (-0.243928, -12.298660)

Steps number: 19
[2]Intersection f2 and f3 [2.000000; 6.000000] ~ (3.243931, 0.924814)

Steps number: 20
[1]Intersection f1 and f3 [-10.000000; -0.000010] ~ (-5.854101, -0.512460)

Steps number: 20
[2]Intersection f1 and f3 [0.000010; 10.000000] ~ (0.854106, 3.512463)
happyjoy@deepspace:~/MIPT/home_c/AC_CW$
```