Московский государственный университет
имени М. В. Ломоносова
Факультет вычислительной математики и кибернетики

Отчет по заданию №6

«Сборка многомодульных программ. Вычисление корней уравнений и определенных интегралов.»

Вариант 3 / 2 / 1

Выполнил:
студент 103 группы
Цыбанов И. А.

Преподаватель:
Кузьменкова Е. А.

Москва
2023

Содержание

1. Постановка задачи

В данном задании необходимо было реализовать метод, который позволяет вычислять площадь плоской фигуры, ограниченной 3 кривыми : $f_1(x) = e^{-x}+3$, $f_2(x) = 2x - 2$, $f_3(x) = \frac{1}{x}$, с точностью $\varepsilon = 0.001$.

Для вычисления определенного интеграла на заданном отрезке использовался метод прямоугольников. Для нахождения точек пересечения кривых использовался метод хорд. Вышеперечисленные методы требовалось реализовать на языке C. Отрезки, на которых должен быть применён метод хорд, требовалось вычислить аналитически.

Вычисление значений кривых в точке нужно было реализовать при помощи языка ассемблера NASM. Сборку многомодульной программы требовалось осуществлять при помощи Makefile.

2. Математическое обоснование

Заметим, что при $x < 0$ $f_3(x) < 0$, а в точке 0 терпит бесконечный разрыв. При этом $f_1(x) > 3$ при любом x. Значит, при $x < 0$ кривые, заданные этими функциями, не пересекаются. Тогда будем рассматривать промежуток $x > 0$.

На этом промежутке $f_1(x)$ монотонно убывает, а $f_2(x)$ монотонно возрастает, причём обе функции непрерывны на промежутке. Значит, они имеют лишь одну точку пересечения на этом промежутке. Аналогично, одну точку пересечения при $x > 0$ имеют функции $f_3(x)$ и $f_2(x)$.

Ясно, что $f_3(x) < 3$ при $x > \frac{1}{3}$, причём она монотонно убывает на промежутке. Так как $f_1(x) > 3$ при любом x, то кривые, заданные этими функциями, имеют не более одной точки пересечения на промежутке. Так как $f_3(x)$ терпит бесконечный разрыв в точке 0, а $f_1(x)$ непрерывна на всей числовой прямой, то они имеют одну точку пересечения на промежутке.

Тогда имеем право обозначить за $x_1 < x_2 < x_3$ точки пересечения кривых $f_1(x)$ с $f_3(x)$, $f_2(x)$ с $f_3(x)$, $f_1(x)$ с $f_2(x)$ соответственно. Тогда следует из графика, что искомая площадь будет равна $S = S_1 - S_2 - S_3$, где $S_1 = \int_{x_1}^{x_3} f_1(x)\,\mathrm{d}x$, $S_2 = \int_{x_1}^{x_2} f_3(x)\,\mathrm{d}x$, $S_3 = \int_{x_2}^{x_3} f_2(x)\,\mathrm{d}x$.

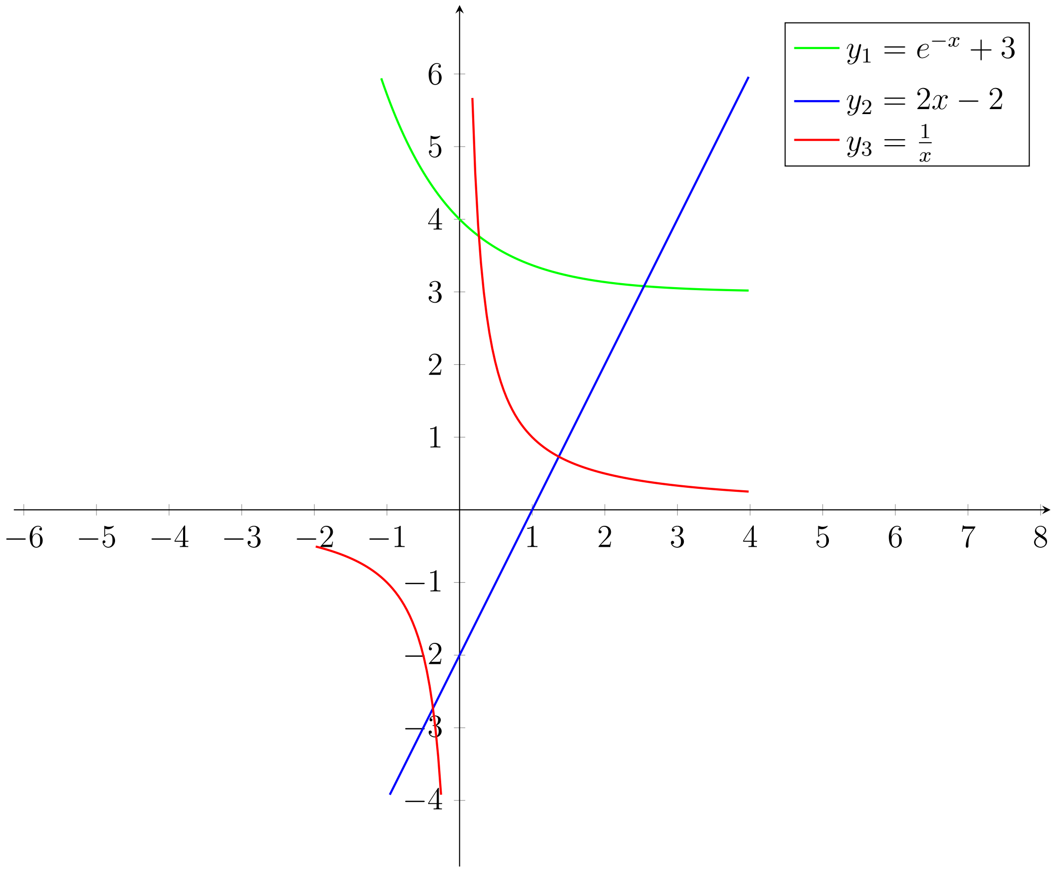


Рис. 1: Плоская фигура, ограниченная графиками заданных функций

1.Обоснование выбора отрезков поиска корней
Положим $F(x) = f(x) - g(x)$ на отрезке $[a, b]$. Тогда, если выполнены условия:

- $F(a) \cdot F(b) < 0$
- $F'(x)$ и $F''(x)$ сохраняют знак на $[a, b]$,

то уравнение $F(x) = 0$ всегда имеет едиственный корень на $[a, b]$ и для его поиска можно воспользоваться методом хорд. [1]

1. $f_1(0.1) - f_3(0.1) = \frac{1}{\sqrt[10]{e}} - 7 < 0$
   $f_1(3) - f_3(3) = \frac{1}{e^3} + \frac{8}{3} > 0$
   При этом на отрезке $[0.1, 3]$ $f_1'(x) - f_3'(x) = -e^{-x} + \frac{1}{x^2} < 0$, а также $f_1''(x) - f_3''(x) = e^{-x} - \frac{1}{2x^3} > 0$.
   Следовательно $f_1(x) - f_3(x)$ удовлетворяет условию на отрезке $[0.1, 3]$.

2. $f_2(0.1) - f_3(0.1) = -11.8 < 0$
   $f_2(3) - f_3(3) = \frac{11}{3} > 0$
   При этом на отрезке $[0.1, 3]$ $f_2'(x) - f_3'(x) = 2 + \frac{1}{x^2} > 0$ (так как каждое из слагаемых на отрезке больше нуля), а также $f_2''(x) - f_3''(x) = -\frac{1}{2x^3} < 0$ (так как каждое из слагаемых на отрезке меньше нуля).
   Следовательно $f_2(x) - f_3(x)$ удовлетворяет условию на отрезке $[0.1, 3]$.

3. $f_1(0.1) - f_2(0.1) = \frac{1}{\sqrt[10]{e}} + \frac{24}{5} > 0$
   $f_1(3) - f_2(3) = \frac{1}{e^3} - 1 < 0$ (так как $e^3 > 10$)
   При этом на отрезке $[0.1, 3]$ $f_1'(x) - f_2'(x) = -e^{-x} - 2 < 0$ (так как каждое из слагаемых на отрезке меньше нуля), а также $f_1''(x) - f_2''(x) = e^{-x} > 0$ (так как каждое из слагаемых на отрезке больше нуля).
   Следовательно $f_1(x) - f_2(x)$ удовлетворяет условию на отрезке $[0.1, 3]$.

2. Обоснование выбора погрешностей ε_1 и ε_2.
При вычислении интеграла методом прямоугольников вычисляются приближения к интегралу $\int_a^b F(x)\,\mathrm{d}x$ вида $I_n = h(F(a_0) + F(a_1) + ... + F(a_{n-1}))$, где $h = \frac{b-a}{n}$, $a_i = a + h(0.5 + i)$. В обосновании выбора отрезков поиска корней было получено, что все точки фигуры лежат на отрезке $[0.1, 3]$. Следовательно, можно заметить, что на этом отрезке $f_1(x) = e^{-x} + 3 < 4 < 10$, $f_2(x) = 2x - 2 \leq 4 < 10$, $f_3(x) = \frac{1}{x} \leq 10$ (все функции не превосходят 10 по модулю).

Далее можно заметить, что при вычислении S_1, S_2, S_3 в качестве пределов интегрирования берутся корни x_1, x_2, x_3, которые в свою очередь были вычислены с погрешностью ε_1, значит h для приближения I_n было вычислено с погрешностью $\frac{2\varepsilon_1}{n}$, из этого следует, что погрешность при высилении приближения I_n составляет меньше чем $10n \cdot \frac{2\varepsilon_1}{n} = 20\varepsilon_1$.

Был применён метод прямоугольников с погрешностью ε_2 3 раза для S_1, S_2 и S_3. Тогда общая погрешность составляет $3 \cdot (20\varepsilon_1 + \varepsilon_2) = 60\varepsilon_1 + 3\varepsilon_2$.

Положим $\varepsilon_1 = \varepsilon_2 = 10^{-6}$, тогда $60\varepsilon_1 + 3\varepsilon_2 = 63 \cdot 10^{-6} = 0.063 \cdot 10^{-3} < 10^{-3} = 0.001 = \varepsilon$. Таким образом, получили искомую погрешность.

3. Результаты экспериментов

В результате проведенных вычислений были найдены координаты точек пересечения кривых, а также площадь полученной фигуры.

Кривые	x	y
1 и 2	2.5395	3.0790
2 и 3	1.3660	0.7320
1 и 3	0.2655	3.7665

Таблица 1: Координаты точек пересечения

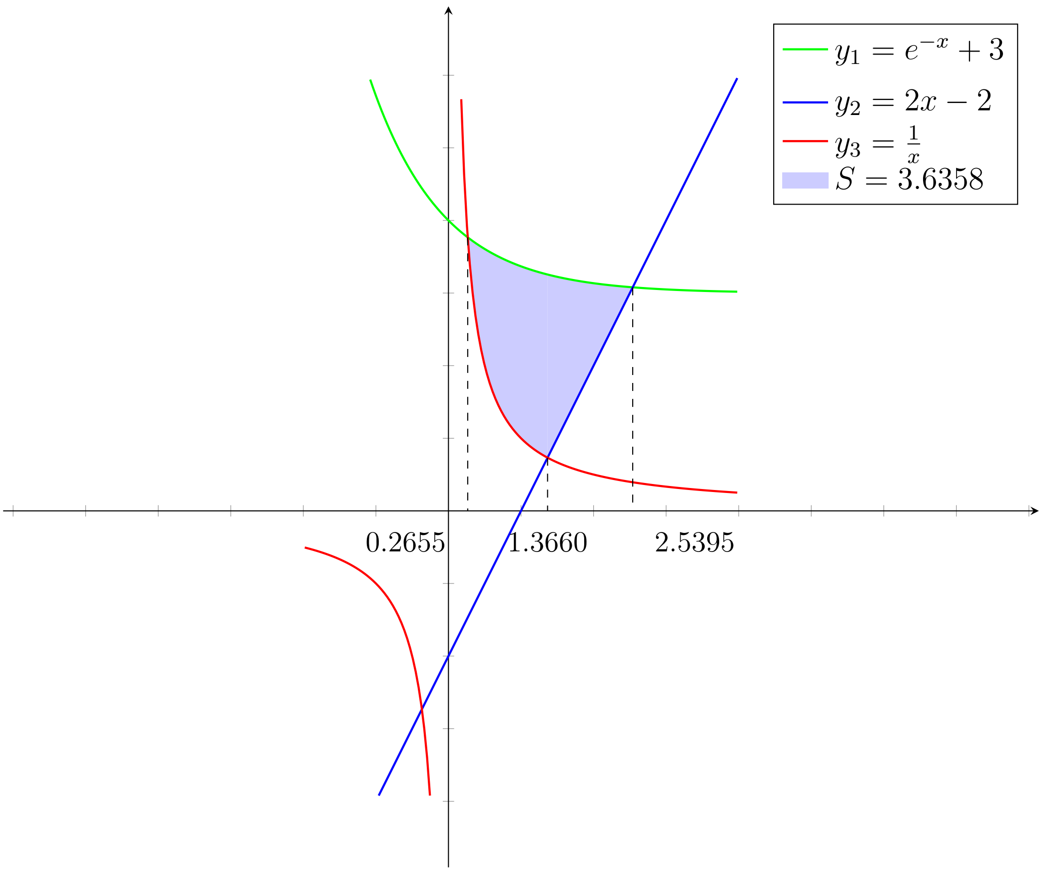


Рис. 2: Плоская фигура, ограниченная графиками заданных функций

4. Структура программы и спецификация функций

- `funcs.asm` - ассемблерный код,состоящий из единственной секции .text, которая состоит из 6 функций, написаннаых по соглашению cdecl. Каждая из функций принимает число x и возвращает f(x) соответственно. В данном модуле программы реализованы все математические функции (f_1, f_2, f_3), которые используются для нахождения исходной площади, и (f_4, f_5, f_6), которые нужны для тестирования Си-функций, реализующих методы прямоугольников и хорд. Список математических функций:
  - $f_1 = e^{-x} + 3$
  - $f_2 = 2x - 2$
  - $f_3 = \frac{1}{x}$
  - $f_4 = x^2 - 2$
  - $f_5 = (x - 2)^3$
  - $f_6 = -2x + 1$
- `integral.c` - код на языке C, состоящий из 2 функций. Данный программный модуль реализует методы прямоугольников и хорд. Список функций данного модуля:
  - `double root (double f1 (double), double f2 (double), double a, double b, double eps1)`; Функция реализует метод хорд нахождения корней уравнения $f1(x) = f2(x)$. Параметры - функции $f1(x)$, $f2(x)$, концы отрезка $[a, b]$, на котором реализуется поиск корня, погрешность eps. Функция работает до того момента, пока расстояние между серединой хорды с одним из концов $[a, b]$ не станет меньше $eps1$. Возвращает середину $[c - eps1, c]$ или $[c, c + eps1]$ в зависимости от стороны приближения к корню (c - середина хорды на текущей итерации функции). Функция также содержет счетчик iter, который хранит количество итераций, понадобившихся для нахождения корня.
  - `double integral (double f (double), double a, double b, double eps2)`; Функция реализует метод прямоугольников нахождения определенного интеграла на отрезке. Параметры - подынтегральная функция $f(x)$, концы отрезка интегрирования $[a, b]$, погрешность $eps2$. Функция работает до того момента, пока не выполнится неравенство: $p|I_n - I_{2n}| < eps$. Возвращает I_{2n} (p - константа Рунге, которая равна $\frac{1}{3}$ для метода прямоугольников).
- `main.c` - код на языке C, состоящий из единственной функции `main`. Задача данного программного модуля - обработка опций командной строки и вызов требуемых функций файла `integral.c`. Список функций:

  - `int main (int argc, char *argv[]);` Функция реализует обработку опций командной строки, проверку их на корректность и вызов соответствующих функций модуля `integral.c`.

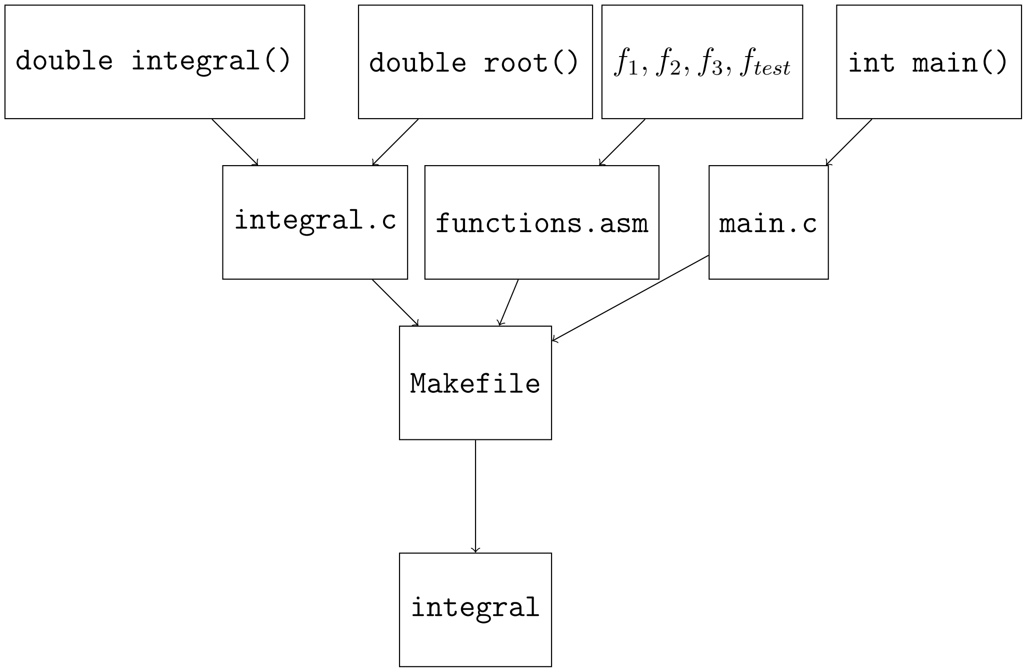
- `Makefile` - файл, в котором содержатся команды, позволяюшие собрать многомодульную программу. `Makefile` содержит цели:

  - `all`: собирает исполняемый файл `integral`
  - `clean`: удаляет все промежуточные (объектные) файлы
  - `test`: запускает файл `integral` в режиме тестирования функций `integral` и `root` (с опциями `-I` и `-R` с заранее заготовленными тестами).

Поддерживаемые функцией `main` опции:

- `-h или -help` - выводит все доступные опции командной строки.
- `-i или -iterations` - выводит количество итераций, за которое были найдены корни.
- `-r или -root` - выводит корни уравнений $f_1(x) - f_2(x) = 0, f_1(x) - f_3(x) = 0, f_2(x) - f_3(x) = 0$
- `-R или -test-root` и `-I или -test-integral` - принимают строку параметр `optarg` в формате `F1:F2:A:B:E:R` для `-R` и `F:A:B:E:R` для `-I` (`F1` и `F2` - номера тестирующих функций, A и B - концы отрезка, `E` - точность, `R` - заранее аналитически посчитанный результат), затем `-I` считает определенный интеграл, `-R` считает корень уравнения. На вывод подается вычисленное значение, абсолютная и относительная погрешности.

Связь модулей



5. Сборка программы (Make-файл)

Для сборки программы был использован Makefile. Его текст приведен ниже:

```
CFLAGS ?= O2 -g
CFLAGS += -std=gnu99
CFLAGS += -Wall -Werror -Wformat-security -Wignored-qualifiers -Winit-self
  -Wswitch-default -Wpointer-arith -Wtype-limits -Wempty-body
  -Wstrict-prototypes -Wold-style-declaration -Wold-style-definition
  -Wmissing-parameter-type -Wmissing-field-initializers -Wnested-externs
  -Wstack-usage=4096 Wmissing-prototypes Wfloat-equal -Wabsolute-value
CFLAGS += -c fsanitize=undefined fsanitize-undefined-trap-on-error
CC = gcc m32 -no-pie fno-pie
LDLIBS = lm
.PHONY: all clean test
all: integral
integral: main.o integral.o funcs.o
     $(CC) $^o $@ $(LDLIBS)
funcs.o: funcs.asm
     nasm -f elf32 $< -o $@
integral.o: integral.c
     $(CC) $(CFLAGS) $< -o $@
main.o: main.c
     $(CC) $(CFLAGS) $< -o $@
clean:
     rm -f main.o integral.o funcs.o integral
test:
     ./integral -R 4:5:1.0:2.0:0.0001:1.564
     ./integral -R 4:6:0.1:2.0:0.0001:1.0
     ./integral -R 5:6:0.1:1.0:0.0001:0.687
     ./integral -I 4:1.0:4.0:0.0001:15.0
     ./integral -I 5:1.0:4.0:0.0001:2.545
     ./integral -I 6:1.0:4.0:0.0001:-12.0
```

- Цель `all` осуществляет сборку в несколько этапов:
  - Цель `funcs.o` осуществляет создание объектного модуля из ассемблерного кода `funcs.asm`.
  - Цель `integral.o` осуществляет создание объектного модуля из C-кода `integral.c`.
  - Цель `main.o` осуществляет создание объектного модуля из C-кода `main.c`.
  - Цель `integral` осуществляет сборку трех ранее созданных объектных фалов в исполняемый файл `integral`.
- Цель `clean` удаляет ранее созданные переходные (объектные) файлы.

- Цель `test` запускает програму в режиме тестирования (с зараенее написанными тестами, с помощью опций `-I` и `-R`).

6. Отладка программы, тестирование функций

Тестирование численных методов производилось с помощью опций `-R` и `-I`. При этом программа запускалась со следующими тестирующими функциями при помощи цели `test`.

1. $f_4(x) = x^2 - 2$
2. $f_5(x) = (x-2)^3$
3. $f_6(x) = -2x + 1$

Функции в параметрах опций `-I` и `-R` имели номера 4, 5, 6 соответственно. Для тестирующей опции `-R` были предложены уравнения $f_4(x) = f_5(x), f_4(x) = f_6(x), f_5(x) = f_6(x)$, для каждого уравнения были выбраны погрешность $\varepsilon = 10^{-4}$ и отрезки $[0, 1.2]$, $[0, 2]$, $[0, 2]$ соответственно. Обоснование и поиск ответа:

1. $f_4(0) - f_5(0) = 6 > 0$, $f_4(1.2) - f_5(1.2) = -\frac{6}{125} < 0$, $f_4'(x) - f_5'(x) = 2x - 3(x-2)^2 < 0$ на $[0, 1.2]$ и $f_4''(x) - f_5''(x) = 2 - 6(x-2) > 0$ на $[0, 1.2]$. Все условия существования и единственности выполнены, тогда: $x^2 - 2 - (x-2)^3 = 0 <=> (x-1)(x-3+\sqrt{3})(x-3-\sqrt{3}) = 0 =>$ на $[0, 1.2]$ $x = 1$.
2. $f_4(0) - f_6(0) = -3 < 0, f_4(2) - f_6(2) = 11 > 0$, $f_4'(x) - f_6'(x) = 2x + 2 > 0$ на $[0, 2]$ и $f_4''(x) - f_6''(x) = 2 > 0$ на $[0, 2]$. Все условия существования и единственности выполнены, тогда: $x^2 + 2x - 3 = 0 <=> (x-1)(x+3) = 0 =>$ на $[0, 2]$ $x = 1$.
3. $f_5(0) - f_6(0) = -9 < 0, f_5(2) - f_6(2) = 3 > 0$, $f_5'(x) - f_6'(x) = 3(x-2)^2 + 2 > 0$ на $[0, 2]$ и $f_5''(x) - f_6''(x) = 6(x-2) < 0$ на $[0, 2]$. Все условия существования и единственности выполнены, тогда: $(x-2)^3 + 2x - 1 = 0 <=> (x-1)(x^2 - 5x + 9) = 0$ - имеет 1 действительный корень $x = 1$.

Номер теста	Аналитическая результат	Результат функции	Абсолютная погрешность	Относительная погрешность
1	1	1.000039	0.000039	0.000039
2	1	0.999965	0.000035	0.000035
3	1	1.000029	0.000029	0.000029

Таблица 2: результаты тестирования функции `root`

.

При тестировании функции `integral` были выбраны следующие отрезки интегрирования: $[0, 3]$, $[5, 6]$, $[10, 15]$ для $f_4(x), f_5(x), f_6(x)$ соответсвенно. Погрешность была выбрана $\varepsilon = 10^{-4}$. Поиск аналитического ответа:

1. $\int_0^3 (x^2 - 2)\,\mathrm{d}x = (\frac{x^3}{3} - 2x)\Big|_0^3 = \frac{27}{3} - 2 * 3 = 3.$
2. $\int_5^6 (x-2)^3\,\mathrm{d}x = \frac{(x-2)^4}{4}\Big|_5^6 = \frac{256}{4} - \frac{81}{4} = \frac{175}{4}.$
3. $\int_{10}^{15} (-2x+1)\,\mathrm{d}x = (-x^2 + x)\Big|_{10}^{15} = -225 + 15 + 100 - 10 = -120.$

Номер теста	Аналитическая результат	Результат функции	Абсолютная погрешность	Относительная погрешность
1	3	2.999912	0.000088	0.000029
2	43.75	43.749966	0.000034	0.000001
3	-120	-120.000000	0.000000	0.000000

Таблица 3: результаты тестирования функции `integral`

.

Для отладки программ на языке ассемблера NASM и C использовалась утилита `gdb`. Для вычисления площади фигуры и абсцисс пересечений кривых использовалась `WolframAlpha`.

7. Анализ допущенных ошибок

В ходе выполнения работы были допущены следующие ошибки:

- Были в неверном порядке определены границы отрезков для интегрирования, вследствие чего итоговая площадь фигуры вычислялась неверно, несмотря на корректность функций `integral` и `root`.
- Опции командной строки считывались небезопасно (при помощи функции `fgets`). Также опции командной строки не проверялись на корректность и соответствие допустимым.
- Для функции `integral` был неправильно определен критерий остановки цикла, вследствие чего программа могла выполняться бесконечно.

Список литературы

[1] Ильин В. А., Садовничий В. А., Сендов Бл. Х. Математический анализ. Т. 1 — Москва: Наука, 1985.