

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 6 / 3 / 3

Выполнил:
студент 102 группы
Грознецкий А. Е.

Преподаватели:
Смирнов Л. М. и Кулагин А. В.

Москва
2023

Содержание

Постановка задачи	2
Математическое обоснование	3
Графики функций	3
Оценка погрешности площади	4
Вычисление погрешностей ε_1 точек пересечения f_1, f_2, f_3	4
Результаты экспериментов	6
Структура программы и спецификация функций	7
Структура файла main.c	7
Структура файла functions.h	7
Структура файла functions.asm	7
Сборка программы (Makefile)	8
Отладка программы, тестирование функций	9
Функция root	9
Функция integral	9
Программа на Си и на Ассемблере	10
Анализ допущенных ошибок	11
Список цитируемой литературы	12

Постановка задачи

Была поставлена задача с заданной точностью $\varepsilon = 0.001$ вычислить площадь плоской фигуры, ограниченной тремя кривыми, заданными уравнениями $y = f_1(x)$, $y = f_2(x)$ и $y = f_3(x)$. Функции f_1 , f_2 , f_3 определяются вариантом, и в частности:

$$f_1(x) = 0.6x + 3,$$

$$f_2(x) = (x - 2)^3 - 1,$$

$$f_3(x) = 3/x.$$

Для решения задачи необходимо написать многомодульную программу на языках Си и Ассемблера, которая должна выполнить следующее:

1. С некоторой точностью ε_1 определить абсциссы точек пересечения кривых, используя предусмотренный вариантом метод приближённого решения уравнения $F(x) = 0$, в частности: метод касательных (Ньютона). Отрезки, в пределах которых программа будет искать точки пересечения, следует определить вручную.
2. Представить площадь заданной фигуры как алгебраическую сумму определенных интегралов и вычислить эти интегралы с некоторой точностью ε_2 по квадратурной формуле, определенной вариантом задания, в частности: по формуле парабол (Симпсона).

Величины ε_1 и ε_2 следует подобрать вручную так, чтобы гарантировалось вычисление площади фигуры с точностью ε .

Математическое обоснование

Графики функций

Проанализируем набор кривых для наилучшего поиска точек пересечения. Нетрудно видеть (рис. 1), что фигура, ограниченная данными кривыми, целиком лежит в квадрате со сторонами $[0, 6]$, $[0, 6]$, однако во избежание деления на ноль при вычислении значений в граничных точках функции $f_3(x) = 3/x$ отступим от нуля на малое значение, так чтобы точки пересечения все еще лежали на новом отрезке. Пусть это будет отрезок $[0.1, 6]$ — именно здесь и будем искать точки пересечения графиков.

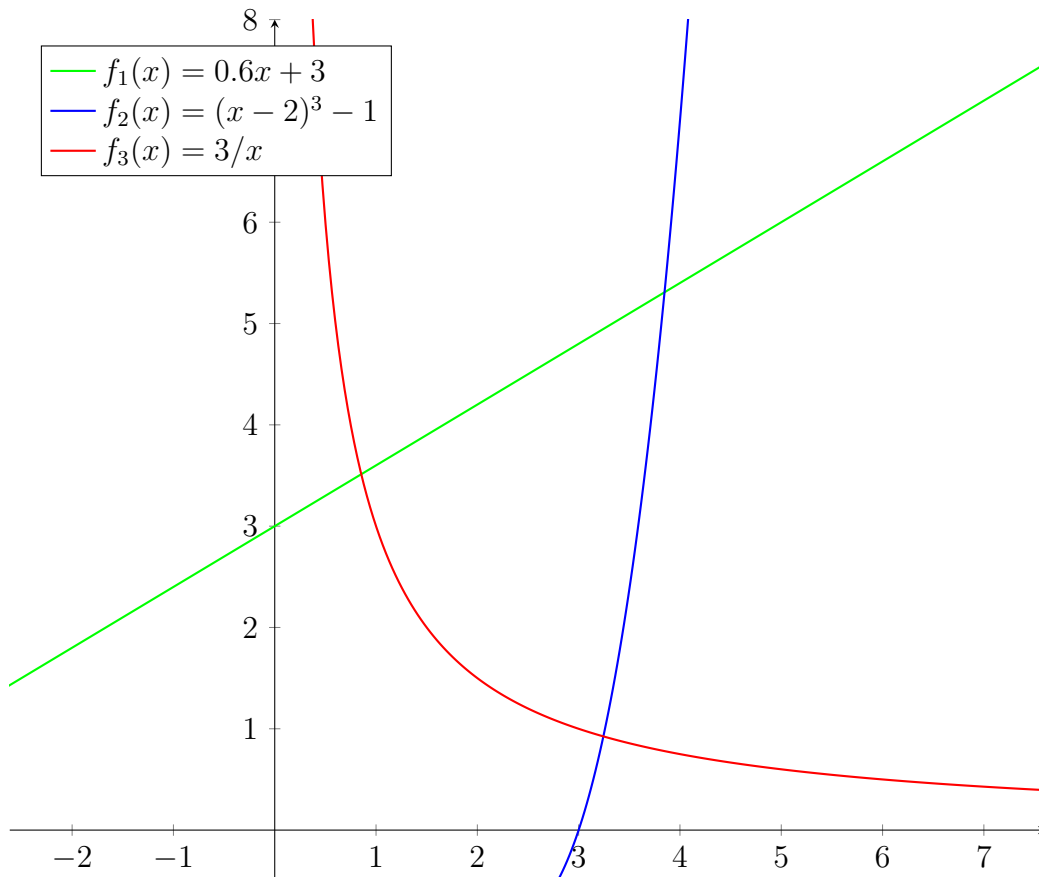


Рис. 1: Плоская фигура, ограниченная графиками заданных уравнений

Определим теперь необходимую точность $\varepsilon_1(\varepsilon)$ нахождения абсцисс точек пересечения графиков и $\varepsilon_2(\varepsilon)$ — точность вычисления параболлической суммы Симпсона.

Площадь фигуры есть алгебраическая сумма трех определенных интегралов, отвечающих площадям под соответствующими кривыми, поэтому вычисление каждого интеграла будем проводить с точностью $\varepsilon/3$. Тогда можно вычислять параболлические суммы с точностью $\varepsilon/6$, а абсциссы точек пересечения с такой точностью, чтобы погрешность их вычисления оказывала влияние на значение интеграла не больше чем $\varepsilon/6$. Имеем $\varepsilon_2 = \varepsilon/6 = 0.001/6$.

Нетрудно заметить (рис. 1), что точки пересечения графиков не совпадают с их экстремальными точками, следовательно можем считать функции монотонными в некоторой окрестности точек пересечения (на самом деле они монотонны не только в окрестности точек пересечения, но это, вообще говоря, неважно).

Оценка погрешности площади

Рассмотрим некоторую функцию, обладающую свойством монотонности в некоторой окрестности некоторых точек и проанализируем влияние возбуждения значений граничных точек на площадь под графиком. В частности, рассмотрим функцию $y = \frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}$, монотонную в окрестности точек $x = -1$ и $x = 1$.

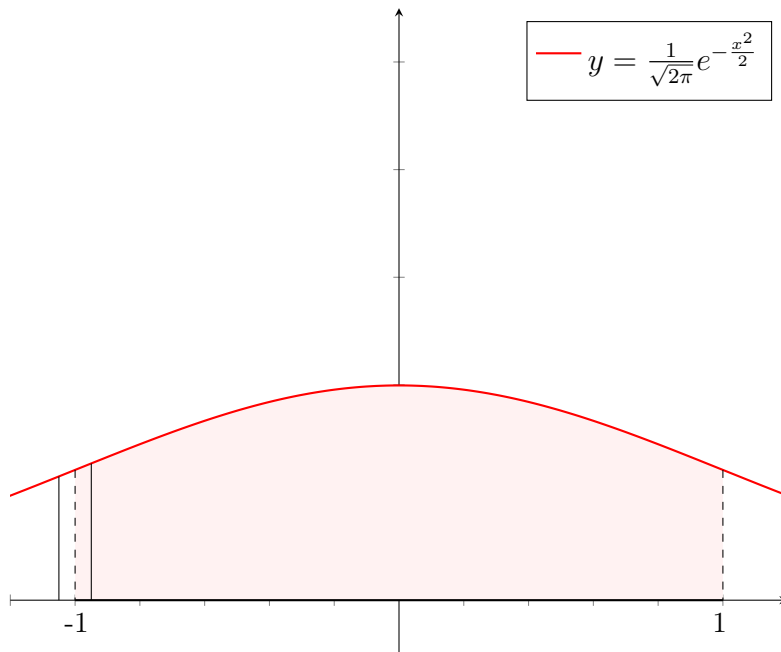


Рис. 2: Некоторая функция, обладающая свойством монотонности в некоторой окрестности некоторых точек

Легко видеть (рис. 2), что изменение положения граничной точки $x_0 = -1$ в пределах $\pm\delta$ изменяет значение площади не более, чем на δM , где M — наибольшее значение на отрезке $[x_0 - \delta, x_0 + \delta]$. Однако для функции $f(x)$, монотонной в δ -окрестности точки x_0 , верно, что наибольшее значение достигается в граничной точке. Таким образом, погрешность вычисления площади $\theta(\delta)$:

$$\theta(\delta) = \delta \cdot \max\{f(x_0 - \delta), f(x_0 + \delta)\}$$

Вычисление погрешностей ε_1 точек пересечения f_1, f_2, f_3

Для начала вычислим аналитически точки пересечения графиков функций f_1, f_2, f_3 .

- $f_1(x) = f_2(x)$ при $x_{1,2} = \frac{\sqrt[3]{(2\sqrt{211} + 13\sqrt{5})^2 + 1}}{\sqrt{5}\sqrt[3]{2\sqrt{211} + 13\sqrt{5}}} + 2$
- $f_1(x) = f_3(x)$ при $x_{1,3} = \frac{3\sqrt{5}}{2} - \frac{5}{2}$
- $f_2(x) = f_3(x)$ при $x_{2,3} = \sqrt{\frac{\sqrt{21}}{2}} + \frac{3}{4} + \frac{3}{2}$

Воспользуемся выведенной формулой для оценки погрешности площади:

$$\theta(\delta) = \delta \cdot \max\{f(x_0 - \delta), f(x_0 + \delta)\}$$

Заметим, что каждая из точек пересечения $x_{1,2}$, $x_{1,3}$, $x_{2,3}$ будет использована для вычисления двух определенных интегралов, поэтому нужно будет выбрать минимальное из двух δ , полученных из уравнения выше для каждой точки. Минимальное δ будет соответствовать той функции, график которой сильнее изменяется в δ -окрестности точки (строго говоря производная в точке по модулю больше). Для каждой точки это функция:

$$x_{1,2}: f_2(x)$$

$$x_{1,3}: f_3(x)$$

$$x_{2,3}: f_2(x)$$

Вообще говоря, в уравнении выше достаточно выполнения неравенства:

$$\theta(\delta) = \delta \cdot \max\{f(x_0 - \delta), f(x_0 + \delta)\} \leq \varepsilon/6$$

Искомое $\varepsilon_1 = \min\{\delta_{1,2}, \delta_{1,3}, \delta_{2,3}\}$, где каждое $\delta_{i,j}$ удовлетворяет соответствующему ему неравенству.

Методом подбора находим подходящие $\delta_{i,j}$: $\delta_{1,2} = 0.00001$, $\delta_{1,3} = 0.00001$, $\delta_{2,3} = 0.0001$. Имеем: $\varepsilon_1 = 0.00001$

Результаты экспериментов

Для получения точек пересечения запустим программу с ключом **points**. Результаты работы программы отобразим ниже (таблица 1):

Кривые	x	y
1 и 2	3.84776	5.30866
1 и 3	0.85410	3.51246
2 и 3	3.24393	0.92480

Таблица 1: Координаты точек пересечения

Также запустим программу с ключом **area**, чтобы получить значение площади фигуры (рис. 3). Полученное значение $S = 7.4884$.

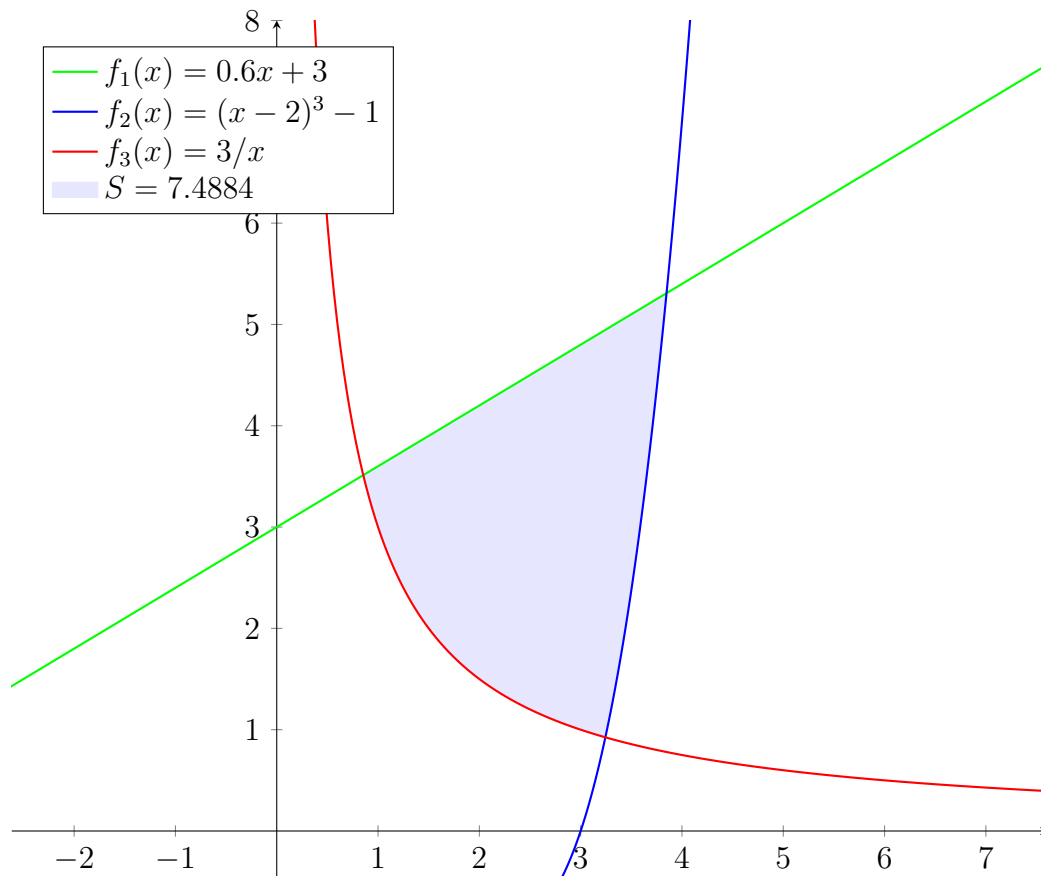


Рис. 3: Плоская фигура, ограниченная графиками заданных уравнений

Структура программы и спецификация функций

Программа состоит из трёх файлов: `main.c`, `functions.h`, `functions.asm`.

Структура файла `main.c`

Файл `main.c` можно разделить на два логически независимых модуля:

1. Функции, отвечающие за взаимодействие с пользователем через командную строку.
2. Функции, выполняющие математические вычисления.

Первый модуль включает в себя функции:

- `void points(int argc, char *argv[]):` *show points of intersection*
- `void iterations(int argc, char *argv[]):` *show count of iterations for calculating intersection points*
- `void area(int argc, char *argv[]):` *show an area of the figure*
- `void test(int argc, char *argv[]):` *test of program functions*
- `void help(int argc, char *argv[]):` *show help information*
- `void no_command(int argc, char *argv[]):` *for unknown commands*

Второй модуль включает в себя функции:

- `double parabolic_integral_sum(function f, double a, double b, int n):` *calculate parabolic integral sum of function $f(x)$ on segment $[a, b]$*
- `double integral(function f, double a, double b, double eps):` *calculate integral of function $f(x)$ on segment $[a, b]$, using `parabolic_integral_sum`*
- `double root(function f, function df, function g, function dg, double a, double b, double eps):` *calculate intersection point of functions: $f(x)$, $g(x)$ on segment $[a, b]$.*

Структура файла `functions.h`

Файл `functions.h` является header файлом, и он содержит лишь заголовки функций `functions.asm`, описанные в следующем разделе.

Структура файла `functions.asm`

Файл `functions.asm` содержит функции f_1 , f_2 , f_3 и их производные с сигнатурой `double (*function)(double x)`.

Сборка программы (Makefile)

Сборка программы происходит с помощью следующего файла Makefile:

```
MAIN_FILE=main.c
OBJ_MAIN_FILE=main.o
EXE_MAIN_FILE=main.out

LIB_FILE=functions.asm
OBJ_LIB_FILE=functions.o

GCC_MAKE_OBJ=gcc -m32 -std=c99 -c ${MAIN_FILE} -o ${OBJ_MAIN_FILE}
NASM_MAKE_OBJ=nasm -f elf32 ${LIB_FILE} -o ${OBJ_LIB_FILE}
GCC_LINK=gcc -m32 ${OBJ_MAIN_FILE} ${OBJ_LIB_FILE} -o ${EXE_MAIN_FILE}
CLEAR_OBJ=rm *.o

all:
    @${GCC_MAKE_OBJ}
    @${NASM_MAKE_OBJ}
    @${GCC_LINK}

clear:
    @${CLEAR_OBJ}
```

Зависимость файлов можно выразить следующей схемой:

```
main.c <- functions.asm
```

то есть файл `main.c` использует функции файла `functions.asm`.

Отладка программы, тестирование функций

Написание программного кода происходило в среде разработки CLion, что позволило избежать ошибок компиляции и отладки программы. Тестирование каждой функции, выполняющей математические вычисления, произведено вручную на трех различных тестах.

Функция root

Функция `root` работает корректно, полученные точки пересечения графиков функций f_1 , f_2 , f_3 действительно вычислены с необходимой точностью:

- $x_{1,2} = \frac{\sqrt[3]{(2\sqrt{211} + 13\sqrt{5})^2 + 1}}{\sqrt{5}\sqrt[3]{2\sqrt{211} + 13\sqrt{5}}} + 2 \approx 3.84776$
- $x_{1,3} = \frac{3\sqrt{5}}{2} - \frac{5}{2} \approx 0.85410$
- $x_{2,3} = \sqrt{\frac{\sqrt{21}}{2}} + \frac{3}{4} + \frac{3}{2} \approx 3.24393$

Значения совпадают с вычисленными!

Функция integral

Функция `integral` работает корректно, полученные значения площади под графиками функций f_1 , f_2 , f_3 на соответствующих сегментах действительно вычислены с необходимой точностью:

- $S_1 = S(f_1, x_{1,3}, x_{1,2}) \approx 13.204$
- $S_2 = S(f_2, x_{2,3}, x_{1,2}) \approx 4.003$
- $S_3 = S(f_3, x_{1,3}, x_{2,3}) \approx 1.712$

Значения совпадают с вычисленными!

Результирующая площадь: $S = S_1 - (S_2 + S_3) = 7.4884$.

Программа на Си и на Ассемблере

Программа содержится в архиве, который приложен к этому отчету.

Анализ допущенных ошибок

Ошибок допущено не было.

Список цитируемой литературы

Вспомогательная литература не была использована.