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 / 1 / 1

Выполнил:
студент 105 группы
Попов А. П.

Преподаватель:
Смирнов А. В.

Москва
2018

Содержание

Постановка задачи **2**

Математическое обоснование **3**

Результаты экспериментов **4**

Структура программы и спецификация функций **5**

Сборка программы (Make-файл) **7**

Отладка программы, тестирование функций **8**

Программа на Си и на Ассемблере **9**

Анализ допущенных ошибок **10**

Список цитируемой литературы **11**

Постановка задачи

Все выше перечисленное должно быть оформлено в виде грамматически и логически связного текста на русском языке. Копирования данного фрагмента из описания задания с подстановкой названия метода и уравнений кривых недостаточно. Необходимо реализовать численные методы, позволяющий вычислять площадь фигуры, которая ограничена тремя кривыми. Для вычисления плащди был использован метод средних прямоугольников, который позволяет с заданной точностью производить вычисления. Но для нахождения площади необходимо также знать коардинаты точек пересечения предложенных графиков, это было сделано методом деления отрезка пополам. Границы отрезков были определены аналитическим методом на основании граффического представления кривых, которые заданы формулами (1), (2), (3).

$$y = e^{-x} + 3 \tag{1}$$

$$y = 2x - 2 \tag{2}$$

$$y = \frac{1}{x} \tag{3}$$

Математическое обоснование

Приведем графики заданных кривых (рис. 1) посредствам графика мы определим границы отрезков, в которых удобно искать точки пересечения графиков, так, например, для поиска точкек пересечения графиков (1) и (2) был выбран отрезок $x \in [2, 3]$, данный выбор очевидно обоснован графически. Аналогично, для графиков (1) и (3) был выбран отрезок $x \in [0, 1]$. Для графиков (2) и (3) - отрезок $x \in [1, 2]$.

Для выбора ε_1 в заданном методе не потребовалось сложных математических вычислений и выводов, так как в методе поиска точки пересечения кривых нужно выбрать $\varepsilon_1 = \varepsilon$, поскольку метод деления отрезка пополам позволяет с заданным ε искать абсциссу точки пересечения графиков с уже заданной точностью ε.

Аналогично выбор ε_2 подразумевает использование уже заданного ε, так метод средних прямоугольников уже дает точность ε при $\varepsilon_2 = \varepsilon$. Это очевидным образом следует из алгоритма поиска площади, так как суммарная погрешность не накапливается и заданный ε_2 влияет лишь на количество точек разбиения примой, что в свою очередь обуславливает и точность найденного значения площади. Данное обоснование также показано в книге [1].

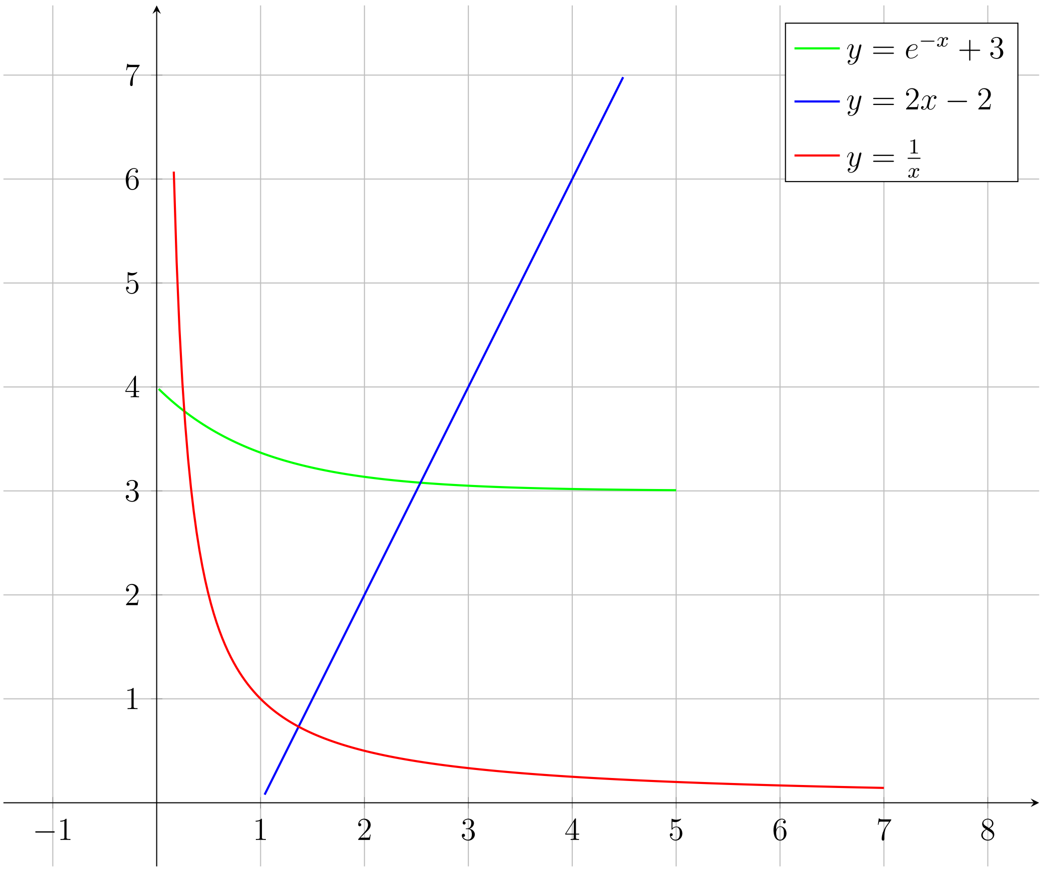


Рис. 1: Плоская фигура, ограниченная графиками заданных уравнений

Результаты экспериментов

Результатом проделанной работы стали числовые значения точек пересечения и искомая площадь, полученные данные представлены в таблице (таблица 1).

Кривые	x	y
1 и 2	2.5395	3.0789
2 и 3	1.3660	0.7321
1 и 3	0.2655	3.7669

Таблица 1: Координаты точек пересечения

Также для иллюстрации результата предложен график заданных функций (1), (2), (3), где область, площадь которой необходимо найти, выделена светло-голубым цветом (рис. 2).

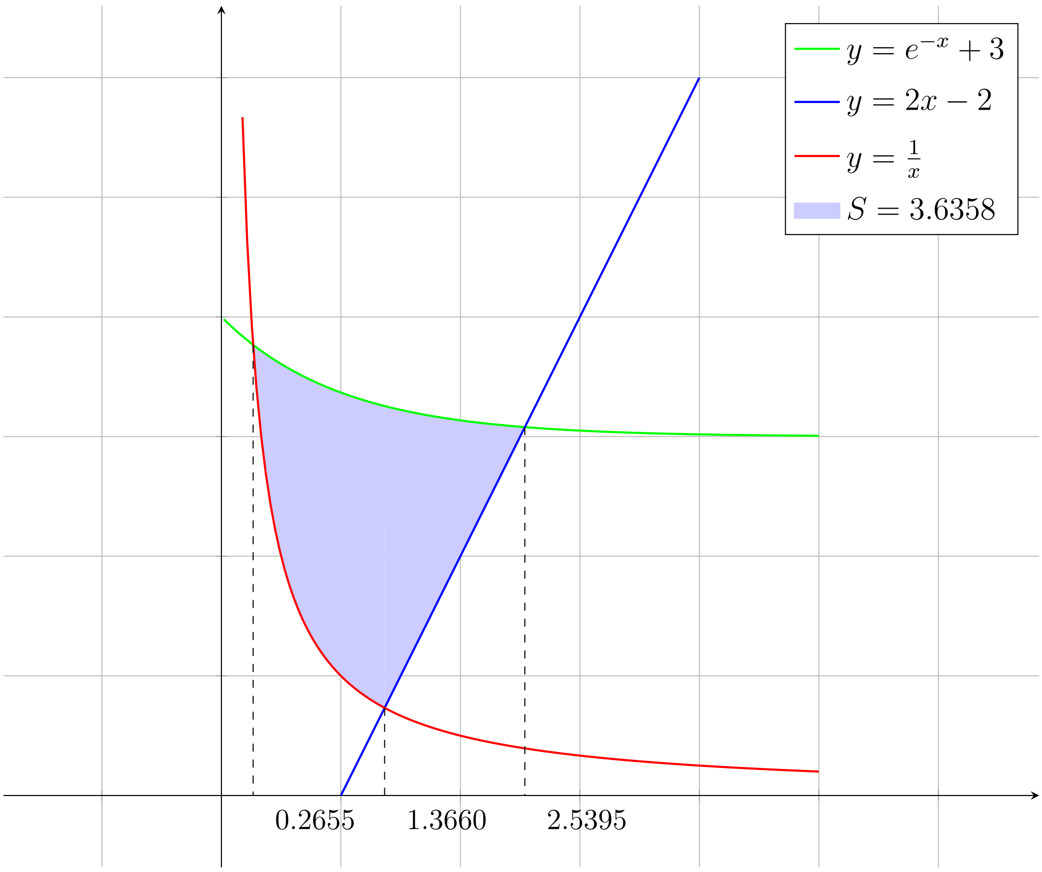
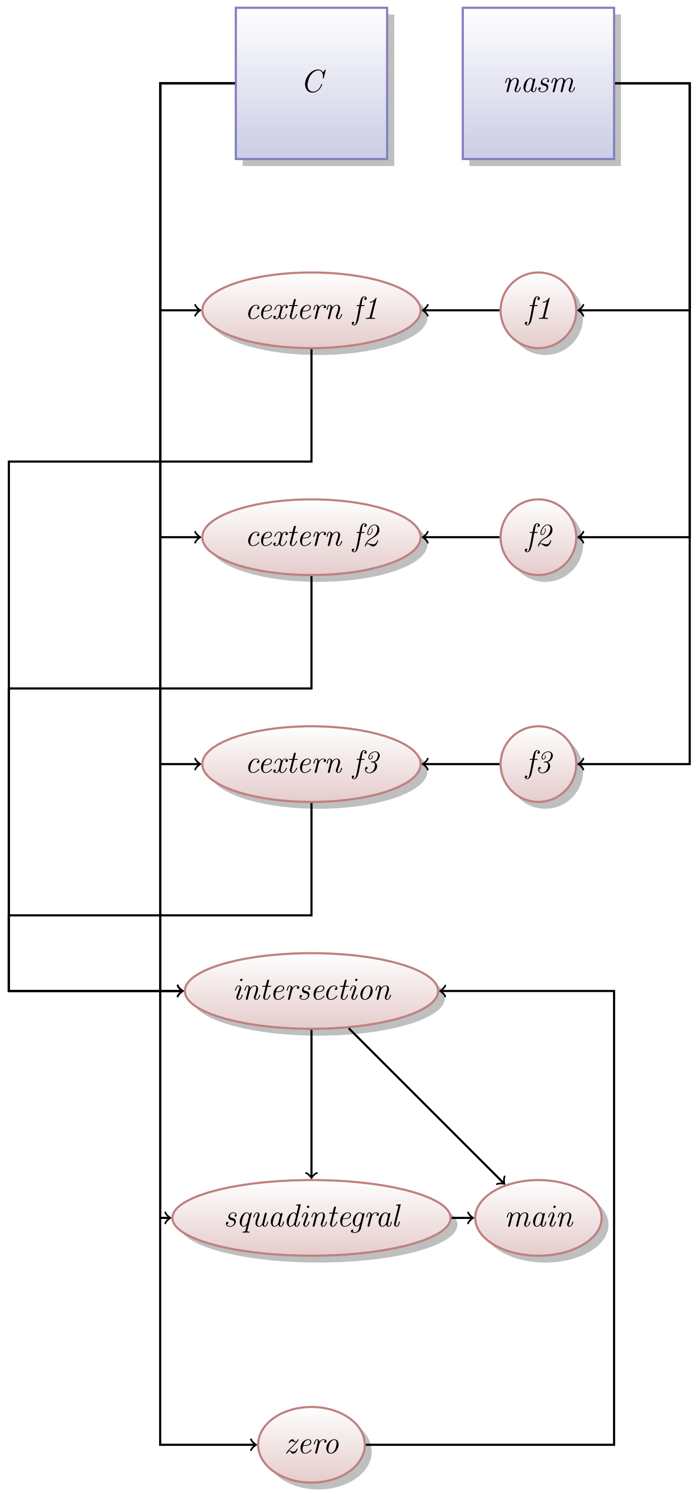


Рис. 2: Плоская фигура, ограниченная графиками заданных уравнений

Структура программы и спецификация функций

Компоненты итоговой программы опишем при помощи диаграммы.



Опишем каждую функцию по отдельности.

1. Раздел NASM:
   1.1. *float f1(float x)*
   1.2. *float f2(float x)*
   1.3. *float f3(float x)*
2. Раздел С:
   2.1. *float intersection(float from , float to , float(*func1)(float) , float(*func2)(float))*
   2.2. *float squadintegral(float from , float to , float(*func1)(float) , float(*func2)(float))*
   2.3. *float zero(float i)*

1.1) Функция, вычисляющая значение выражения $y = e^{-x}+3$, принимает значение x.

1.2) Функция, вычисляющая значение выражения $y = 2x-2$, принимает значение x.

1.3) Функция, вычисляющая значение выражения $y = \frac{1}{x}$, принимает значение x.

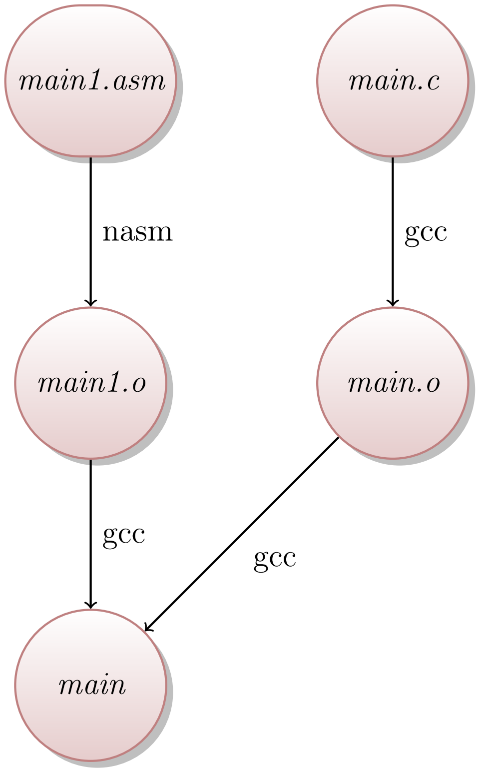
2.1) Функция, вычисляющая абсциссу точки пересечения переданных в качестве параметра функций посредствам алгоритма деления отрезка пополам, также в качестве параметра принимает границы на которых будет производиться поиск, точность производимых действий задается константой.

2.2) Функция подсчета площади между функциями переданными в качестве параметров посредствам алгоритма средних прямоугольников. Границы вычисления площади передаются в качестве параметров. Точность с которой производятся вычисления заданы константой.

2.3) Функция, необходмая для работы режима отладки, при любых входных параметрах она возвращает значение 0.

Сборка программы (Make-файл)

В диаграмме ниже описаны связи компонентов программы в Make-файле.



Текст Make-файла:

```
all: main

main: main.o main1.o
	gcc -w -m32 main.o main1.o -o main

main.o: main.c
	gcc -std=c99 -m32 -Wall -Werror -c main.c

main1.o: main1.asm
	nasm -f macho –prefix _ main1.asm

clean:
	rm -f *.o main
```

Отладка программы, тестирование функций

Приведем тесты на которых производилась отладка программы и ее отдельных компанентов.

1) Функция вычисления $y = e^{-x} + 3$ (1.1)

x	y
-1	5.7183
0	4
1	3.3679

2) Функция вычисления $y = 2x - 2$ (1.2)

x	y
1	0
2	2
-1	-4

3) Функция вычисления $y = \frac{1}{x}$ (1.3)

x	y
1	1
2	0.5
3	0.3333

4) Отладка функции поиска точек пересечения графиков (2.1)

f1	f2	Нижняя граница	Верхняя граница	x	Аналит. реш-е.
x	$\frac{1}{x}$	0.5	2	1.0000	1
x	x^2	0	2	1.0000	1
x^2	$sqrt(x)$	0.5	2	1.0000	1

5) Отладка функции расчета площади между графиками функций (2.2)

f1	f2	Нижняя граница	Верхняя граница	S	Аналит. реш-е.
x	$const = 0$	0	2	1.9998	2
x	x^2	0	1	0.1667	$\frac{1}{6}$
x^2	$sqrt(x)$	0	1	0.3333	$\frac{1}{3}$

Программа на Си и на Ассемблере

Исходные тексты программ содержатся в архиве, который приложен к отчету. Архив зашифрован. Так как некоторые почтовые сервисы блокируют сообщения с архивами. Пароль от архива "105".

Анализ допущенных ошибок

В процессе выполнения задания приходилось сталкиваться с трудностями, так, например, для того, чтобы программа собиралась на MacOS необходимо было добавить специальный ключь для линковщика gcc, который позволяет избежать проблему различного определения имен функций.

Также возникала проблема с реализацией функции расчета e^{-x}, так как стандартные настройки округления в процессоре x87 не позволяли вычилсять это выражения с достаточной точностью было принято решение изменить правила округления, что позволило существенно увеличить точность вычислений.

Список литературы

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