Московский государственный
университетимени М. В. Ломоносова
Факультет вычислительной математики и
кибернетики

Отчет по заданию №6

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

Вариант 10 / 1 / 1

Выполнила:
Хорошилова Е.Д. 106 группа
Преподаватели:
Корухова Л.С., Манушин Д.В.

Москва
2022

Содержание

Постановка задачи

С помощью метода приближенных вычислений найти площадь плоской фигуры, ограниченной тремя кривыми:

$$f_1 = 1 + \frac{4}{x^2 + 1}, \qquad f_2 = 2^{-x}, \qquad f_3 = x^3.$$

Для поиска корней использован метод деления отрезка пополам, для вычисления интеграла (площади искомой фигуры) используется метод прямоугольников.

Вычисленные корни являются абсциссами вершин фигуры.

Аналитически вычислили отрезок для нахождения корней (см. Математическое обоснование).

Математическое обоснование

Функция $\boldsymbol{f_1(x)}$ существует при любых x, чётная, монотонно возрастает при отрицательных x, монотонно убывает при положительных x , область значений от 1 до 5.

Функция $\boldsymbol{f_2(x)}$ существует при любых x и монотонно убывает, область значений от 0 до $+\infty$.

Функция $\boldsymbol{f_3(x)}$ существует при любых x и монотонно возрастает, область значений от $-\infty$ до $+\infty$.

Рассмотрим функцию $\mathbf{F_1(x) = f_1(x) - f_2(x),\ F_1(10) > 0,\ F_1(-2) < 0.}$

Рассмотрим функцию $\mathbf{F_2(x) = f_2(x) - f_3(x),\ F2(10) < 0,\ F_2(-2) > 0.}$

Рассмотрим функцию $\mathbf{F_3(x) = f_3(x) - f_1(x),\ F_3(10) > 0,\ F_3(-2) < 0.}$

Значения функций F имеют различные знаки в точках 10 и -2, следовательно, можем выбрать отрезок (-2,10) для поиска корней вследствие непрерывности функций F.

Будем производить вычисление с точностью ε_1 для поиска абсциссы точки пересечения кривых и с точностью ε_2 для поиска интеграла.

Пусть $\boldsymbol{x_1, x_2, x_3}$ – корни функций F_1, F_2, F_3. Тогда искомый интеграл фигуры может быть найден по формуле $\int_{x_1}^{x_3} \boldsymbol{f_1(x)} - \int_{x_1}^{x_2} \boldsymbol{f_2(x)} - \int_{x_2}^{x_3} \boldsymbol{f_3(x)}$.

Поиск корней в нашей программе производится с точность $\boldsymbol{\varepsilon_1}$. Для поиска интеграла используется метод прямоугольников, который обеспечивает точность $\boldsymbol{\varepsilon_2 = 0.0001}$ (пособие [1]). Для уменьшения ошибки, вносимой погрешностью вычислений границ для интегрирования на концах отрезка, возьмем $\boldsymbol{\varepsilon_1 \ll \varepsilon_2}$, в данном случае $\boldsymbol{\varepsilon_1 = 0.1 * \varepsilon_2 = 0.00001}$.

Данные методы взяты из книги [2].

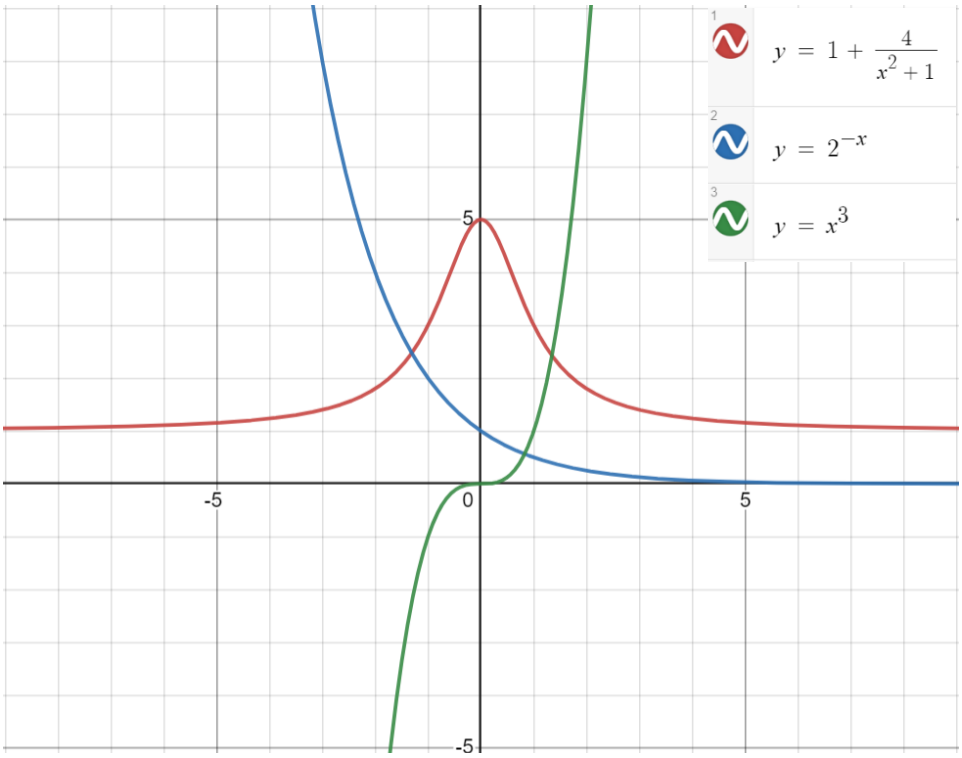

Рис. 1: Плоская фигура, ограниченная графиками заданных уравнений

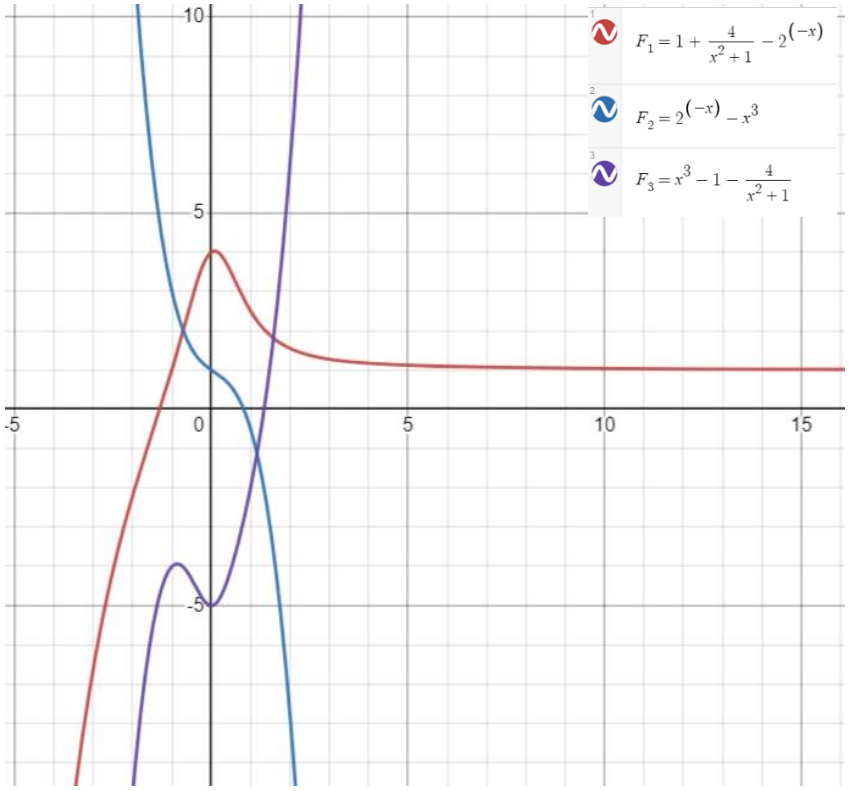

Рис. 2: Графики разности функций для определения знакопеременности и отрезков, содержащих корни.

Результаты экспериментов

Вычисление координат точек пересечения кривых (погрешность **EPS = 0.001**):

Кривые	*x*	*y*
1 и 2	1.34365	2.425819
2 и 3	0.826218	0.564006
1 и 3	-1.30786	2.47574

Таблица 1: Координаты точек пересечения

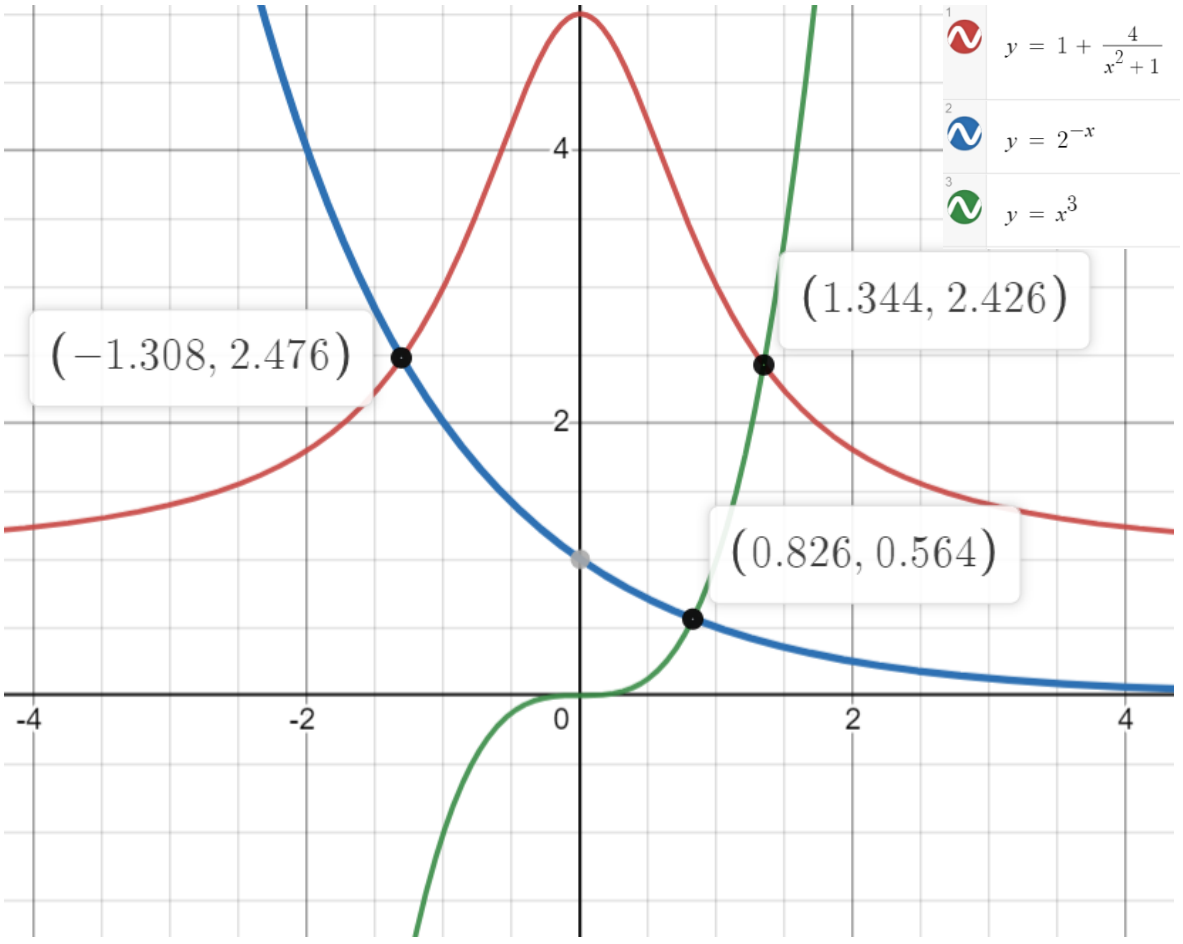

Рис. 3: Плоская фигура, ограниченная графиками заданных уравнений. Координаты точек пересечения кривых.

Искомая площадь S = 6.5911, погрешность 0.012.

Структура программы и спецификация функций

Список модулей:

main.c – основная программа, которая производит основные вычисления.

functions. asm – модуль, содержащий функции f_1, f_2, f_3 на языке Ассемблера NASM (Архитектура IA-32).

Список функций:

main.c:

```
double root (double (* f1) (double x), double (* f2) (double x), double x_left, double x_right, double eps)
```

Функция, вычисляющая точки пересечения двух функций на подаваемом отрезке с точностью ε.

```
double integral (double (* f)(double x), double a, double b, double eps2)
```

Функция, вычисляющая интеграл функции на подаваемом отрезке с точностью ε.

```
int test_root (double (* root) (double (* f1) (double x), double (* f2) (double x), double x_left, double x_right, double eps))
```

Функция, тестирующая root (3 теста).

```
int test_integral (double (*integral) (double (* f) (double x), double a, double b, double eps2))
```

Функция, тестирующая integral (3 теста).

```
double square_of_figure (double (* f1) (double x), double (* f2) (double x), double (* f3) (double x))
```

Функция, вычисляющая площадь фигуры, заключенной между тремя графиками.

```
void testik (int n)
```

Функция, которая тестирует определенный тест по ключу -test и номеру, который подается через n. Например, -test 5.

functions. asm:

`double` $f_1\ (double\ x)$ – функция f_1.

`double` $f_2\ (double\ x)$ – функция f_2.

`double` $f_3\ (double\ x)$ – функция f_3.

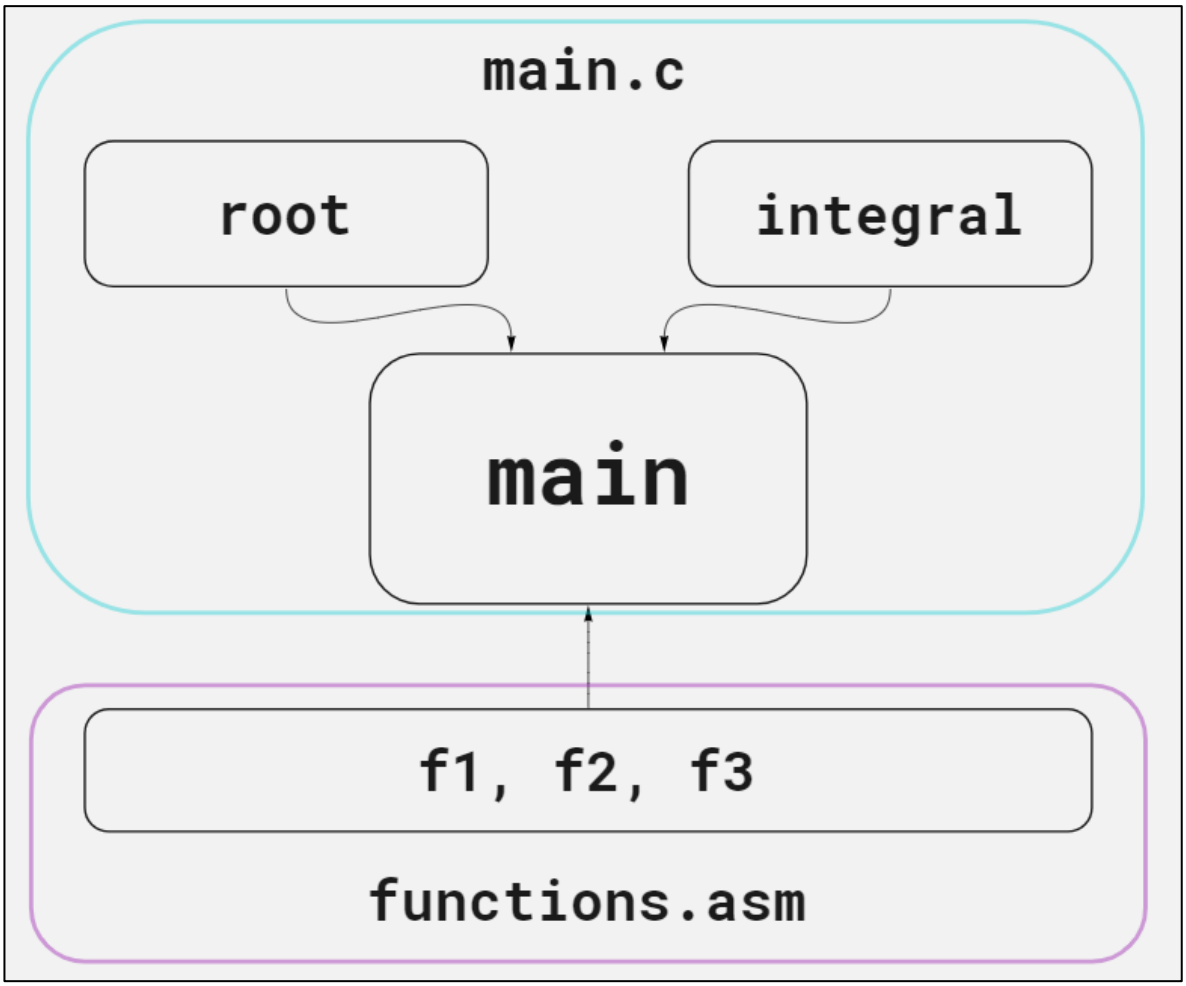

Рисунок 4: связь между компонентами программы.

Сборка программы (Make-файл)

Рисунок 5: Сборка программы.

Текст Make-файла:

```
all: main
clean:
    rm main ./*.o
main.o: main.c
    gcc -c -o main.o main.c -m32 -lm
functions.o: functions.asm
    nasm -f elf32 -o functions.o functions.asm
main: main.o functions.o
    gcc -o main main.o functions.o -m32
```

Информация о методах работы с Make-файлом была взята из [3].

Отладка программы, тестирование функций

Возьмем 3 данных функции и протестируем возвращаемые значение этих функций для нескольких точек.

$$f_1 = 1 + \frac{4}{x^2 + 1}, \qquad f_2 = 2^{-x}, \qquad f_3 = x^3.$$

Убедимся, что возвращаемое значение лежит в пределах допустимой погрешности ε_1 с результатом, рассчитанным аналитически.

	Аргумент 1	Значение	Аргумент 2	Значение	Аргумент 3	Значение
f_1	0	5	1.7	2.028	-2.8	1.452
f_2	0	1	1.7	0.308	-2.8	6.964
f_3	0	0	1.7	4.913	-2.8	-21.952

Таблица 2: Вычисление значений функций в точках.

Подберем отрезки, на которых выполняются условия нахождения корня методом деления отрезка пополам. На данных значениях проверим возвращаемое значение функции root, сравнив значения с ожидаемыми.

Кривые	Действительный корень	Найденный корень
1 и 2	1.34365	1.343656
2 и 3	0.826218	0.826222
1 и 3	-1.30786	-1.307857

Таблица 3: Координаты точек пересечения

Протестируем функцию интеграла аналогичным образом, на данных функциях f_1, f_2, f_3.

Функция	Отрезок	Аналитически вычисленный интеграл	Возвращенное значение функции
f_1	[3, 10]	7.88833	7.8883
f_2	[0, 1]	0.25	0.249969
f_3	[-2, 5]	5.7257	5.72567

Таблица 4: Получение значений интегралов.

Из полученных значений можно сделать вывод, что значения функций, корни и интегралы, вычисленные аналитически и полученные в результате работы программы, совпали с требуемой точностью.

Проверку с табличными значениями можно запустить с параметром **–test** n, где n – номер теста от 1 до 6.

Программа на Си и на Ассемблере

В приложенном к отчету архиву находится исходный код программы в файле **main.c.** Функции, реализованные на языке ассемблер, находятся в файле **functions.asm. Makefile** проекта также присутствует в архиве.

Zip-архив защищен паролем «123», для безопасности пересылки.

Анализ допущенных ошибок

1. Ошибки при работе с ***main.c:***
    a. При введении номера несуществующего теста программа выдавала segmentation fault.
    b. При поиске корня программа «залипала на крайнем значении подаваемого отрезка значений абсцисс» (неправильно высчитывался центр между двумя текущими абсциссами)
2. Ошибки при работе с ***functions.asm:***
    a. Ошибка при работе с вещественными константами
    b. Ошибка при обработке переданных значений функции по соглашению о вызове.
3. Ошибки при работе с ***Makefile:***
    a. Ошибка несовместимости архитектур объектных файлов
    b. Попытка сборки 32-битных программ без установленного 32-битного компилятора
4. Ошибки при работе с ***git:***
    a. Выдавал ошибку к сгенерированной паре ключей.

Список литературы

[1] Е.А. Кузьменкова, В.С. Махнычев, В.А. Падарян. Семинары по курсу «Архитектура ЭВМ и язык ассемблера»: учебно-методическое пособие. Часть 1. — Издание 2-е, дополненное. М. Издательский отдел факультета ВМиК МГУ им. М.В. Ломоносова (лицензия ИД № 05899 от 24.09.2001); МАКС Пресс, 2014. – 80 с.

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

[3] https://habr.com/ru/post/155201/ - статья “Makefile для самых маленьких”.

[4] Трифонов Н.П., Пильщиков В.Н. Задания практикума на ЭВМ (1 курс). Учебное пособие, 2-е исправленное издание. — М.: МГУ, 2001. — 32 с.

[5] А.Н. Степанов. Архитектура вычислительных систем и компьютерных сетей. Издательство: Питер, 2007 г. — 512 стр.