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 1

Выполнил:
студент 106 группы
Широков А. П.

Преподаватели:
Корухова Л. С.
Соловьев М. А.

Москва
2018

Содержание

Постановка задачи

С заданной точностью ε вычислить площадь плоской фигуры, ограниченной тремя кривыми, уравнения которых $y = 2^x + 1$, $y = x^5$ и $y = \frac{1-x}{3}$.

- С некоторой точностью ε_1 вычислить абсциссы точек пересечения кривых, используя комбинированный метод приближенного решения уравнения $F(x) = 0$. Отрезки, где программа будет искать точки пересечения, и где применим используемый метод, следует определить вручную.

- Представить площадь заданной фигуры как алгебраическую сумму определенных интегралов и вычислить эти интегралы с некоторой точностью ε_2 по квадратурной формуле Симпсона.

Величины ε_1 и ε_2 подобрать вручную так, чтобы гарантировалось вычисление площади фигуры с точностью $\varepsilon = 0.001$.

Математическое обоснование

Для поиска нужных интервалов пересечений кривых удобно было построить вспомогательные ф-ии (рис. 1, 2, 3) и выбрать отрезки, на которых содержится пересечение кривой с осью OX и сохраняется монотонность и выпуклость.

За ε_1 (погрешность вычисления корней уравнений), ε_2(погрешность вычисления интегралов) я принял значение, равное 0.0001. При подсчёте каждого из трёх интегралов могут возникнуть ошибки в вычислении граничных точек отрезка(максимальная из них – e_1), а также ошибки в вычислении самого интеграла(e_2). Пусть a, b – верные границы интегрирования, тогда, разбив получившийся интеграл на сумму трёх интегралов, получим: $I' = A+I+B+e_2$, где I – верно вычисленный интеграл, A, B – интегралы-погрешности, I' – полученный интеграл. Возьмём такие наименьшие числа da, db, что $da(db) \geq |f(x)|$, $\forall x$ из ε_1-окрестности точки a(b). Получим: $|I' - I| = |A+B+e_2| \leq |A|+|B|+|e_2| \leq$ (A', B' – интегралы от модулей подынтегральных функций A, B) $\leq |A'|+|B'|+|e_2| \leq |(da+db)*e_1| + \varepsilon_2 \leq \varepsilon_1 * (da+db) + \varepsilon_2 \leq \varepsilon$. Так как da, db выбирались наименьшими, то при ε_1, $\varepsilon_2 = 0.0001$, $\varepsilon = 0.001$, получим верное неравенство.

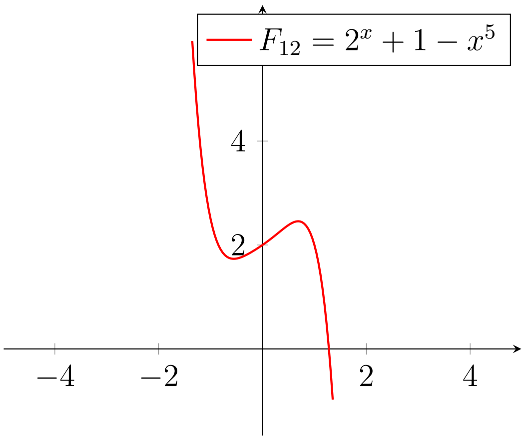

Рис. 1: Вспомогательная кривая для поиска пересечения f_1 и f_2

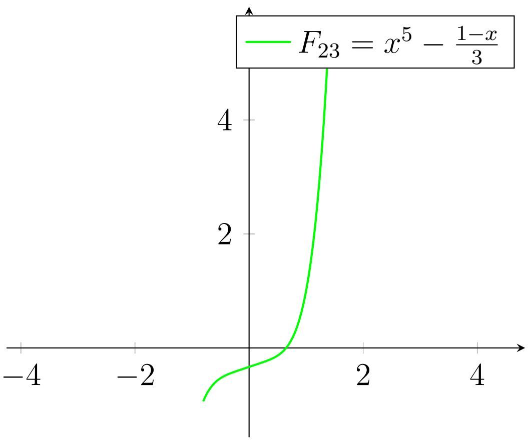

Рис. 2: Вспомогательная кривая для поиска пересечения f_2 и f_3

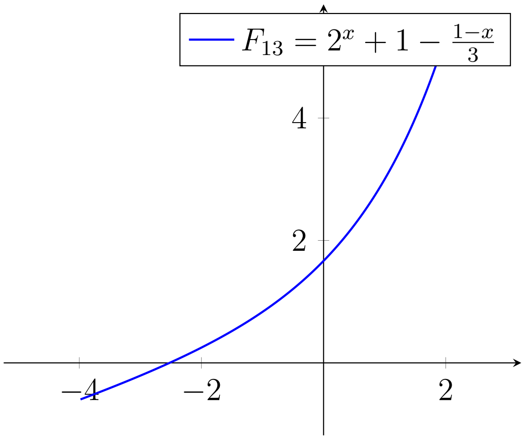

Рис. 3: Вспомогательная кривая для поиска пересечения f_1 и f_3

Результаты экспериментов

В ходе вычислений я получил следующие результаты: координаты точек пересечения (таблица 1) и площадь полученной фигуры (рис. 4).

Кривые	x	y
1 и 2	1.279351	3.427297
2 и 3	0.650537	0.116509
1 и 3	-2.527982	1.173381

Таблица 1: Координаты точек пересечения

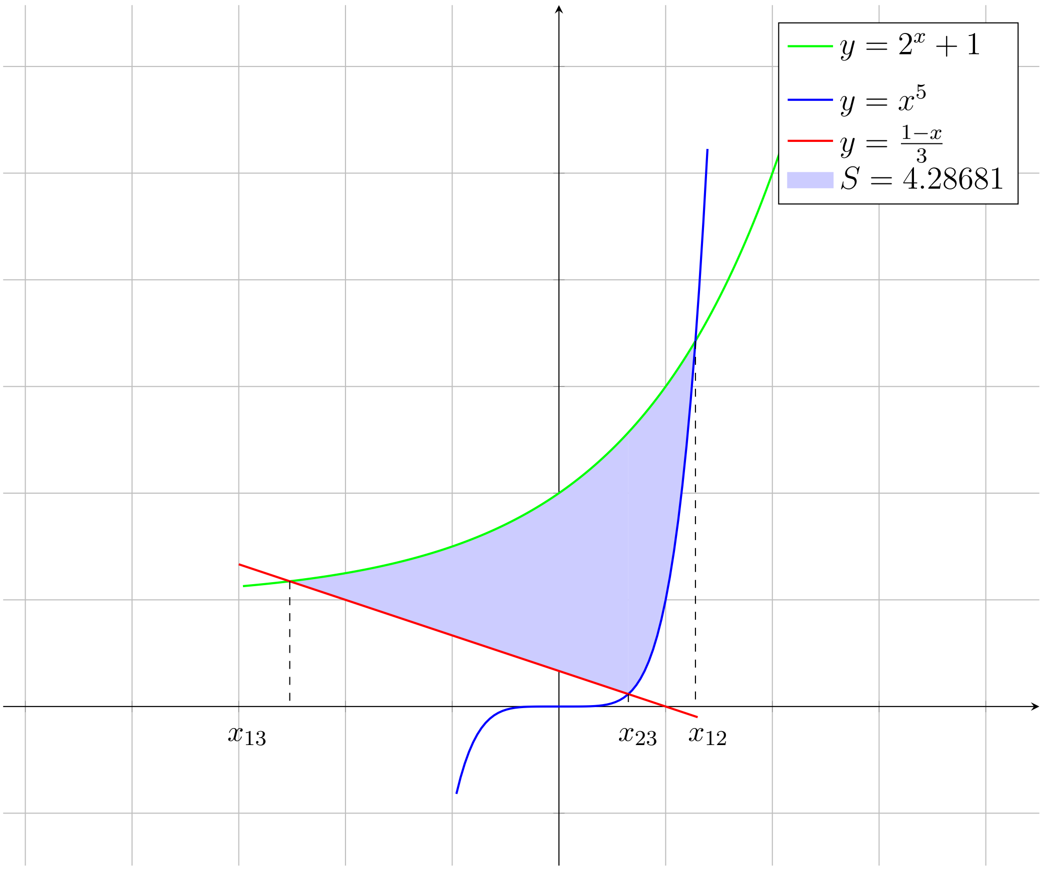

Рис. 4: Плоская фигура, ограниченная графиками заданных уравнений

Структура программы и спецификация функций

Программа состоит из двух модулей: *main.c* и *prak.asm*. В первом описаны методы вычисления интегралов и отыскания корней уравнения, во втором – функции.

- `int main(int argc, char *argv[])`

  Стандартная функция main с двумя параметрами *argc* и *argv*, через которые из командной строки подаются ключи для тестирования программы. Все ключи компиляции можно увидеть по команде $-help$.

- `double root(double (*f)(double), double (*g)(double), double (*pf)(double), double (*pg)(double), double a, double b, double eps)`

  Функция поиска точки пересечения функций f и g, с производными pf и pg на отрезке $[a, b]$ с точностью eps.

- `double integral(double (*f)(double), double a, double b, double eps)`

  Функция вычисления определённого интеграла от функции f на отрезке $[a, b]$ с точностью *eps*.

- `double calculation(double (*f)(double), double border, double *c4, double *c2, int n, double a, double b)`

  Вычисление I_{2n}-ого для функции f на отрезке $[a, b]$, по предыдущему разбиению $I_n = \frac{b-a}{3n} * (y_0 + 2(y_2 + y_4 + ... + y_{n-2}) + 4(y_3 + ... + y_{n-1}) + y_n)$, где $y_0 + y_n = border$, $2(y_2 + y_4 + ... + y_{n-2}) = c_2$, $4(y_3 + ... + y_{n-1}) + y_n) = 2c_4$

- `bool type(double (*f)(double), double (*g)(double), double a, double b)`

  Определение случая комбинированного метода для f и g на отрезке на $[a, b]$. Случай 0 — метод хорд справа, касательных слева; случай 1 — наоборот.

- `double tangent(double (*f)(double), double (*g)(double), double (*pf)(double), double (*pg)(double), double a, double b, bool t)`

  Обновление границы интервала по методу касательных (случай t) для f и g, с производными pf и pg на отрезке на $[a, b]$.

- `double chord(double (*f)(double), double (*g)(double), double a, double b)`

  Обновление границы интервала по методу хорд для f и g на отрезке на $[a, b]$.

Сборка программы (Make-файл)

Так как программа состоит из двух файлов, содержащих функции, (main.c использует функции, описанные в prak.asm) необходимо было написать Makefile.

Текст Makefile (Linux):

```
all: program clean
program: a.o b.o
    -m32 -o program a.o b.o
a.o: main.c
    -m32 -std=c99 -o a.o -c main.c
b.o: prak.asm
    nasm -f elf32 -o b.o prak.asm
clean:
    rm a.o
    rm b.o
```

Отладка программы, тестирование функций

Для отладки были добавлены ключи командной строки $integralTest$, $rootTest$, с помощью которых можно было вывести значение, ф-ий $integral$ и $root$ соответсвенно, от передаваемых через консоль аргументов. Кроме этого использовались и отладочные выводы значений функций $f1$, $f2$, $f3$. Ответ, возвращаемый на удовлетворяющих условиям тестах, сошёлся с посчитанным аналитически, а также с полученным в результате работы программы.

Программа на Си и на Ассемблере

Исходные тексты программы (main.c, prak.asm), а также Makefile хранятся в приложенном к отчёту архиве.

Анализ допущенных ошибок

В ходе работы над данным заданием я неправильно считывал переменные типа $double$, что приводило к $segmentation\ fault$.

Список литературы

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