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

Отчёт по заданию №6

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

Вариант 1 / 5 / 2

Выполнил:
студент 105 группы
Космынин И. Н.

Преподаватели:
Гуляев А. В.
Русол А. В.

Москва
2024

Содержание

Постановка задачи

Реализовать численный метод, позволяющий вычислить площадь плоской фигуры, ограниченной тремя кривыми, заданными функциями $f_1(x) = 2^x + 1$, $f_2(x) = x^5$, $f_3(x) = \frac{1-x}{3}$. Для решения этой задачи используется метод трапеций при вычислении определённого интграла и реализуются два метода для приближённого вычисления абсцисс вершин фигуры (корней уравнений): метод хорд и метод деления отрезка пополам. При этом необходимо аналитически определить отрезки, на которых производится поиск корней.

Программа должна иметь модуль написанный на языке Ассемблера, вычисляющий значение функций f_1, f_2 и f_3 в вещественных числах при помощи сопроцессора x87. Соответствующие функции реализуются с соглашением `cdecl` для дальнейшего использования в Си-коде.

Математическое обоснование

Для анализа предложенных функций построен график:

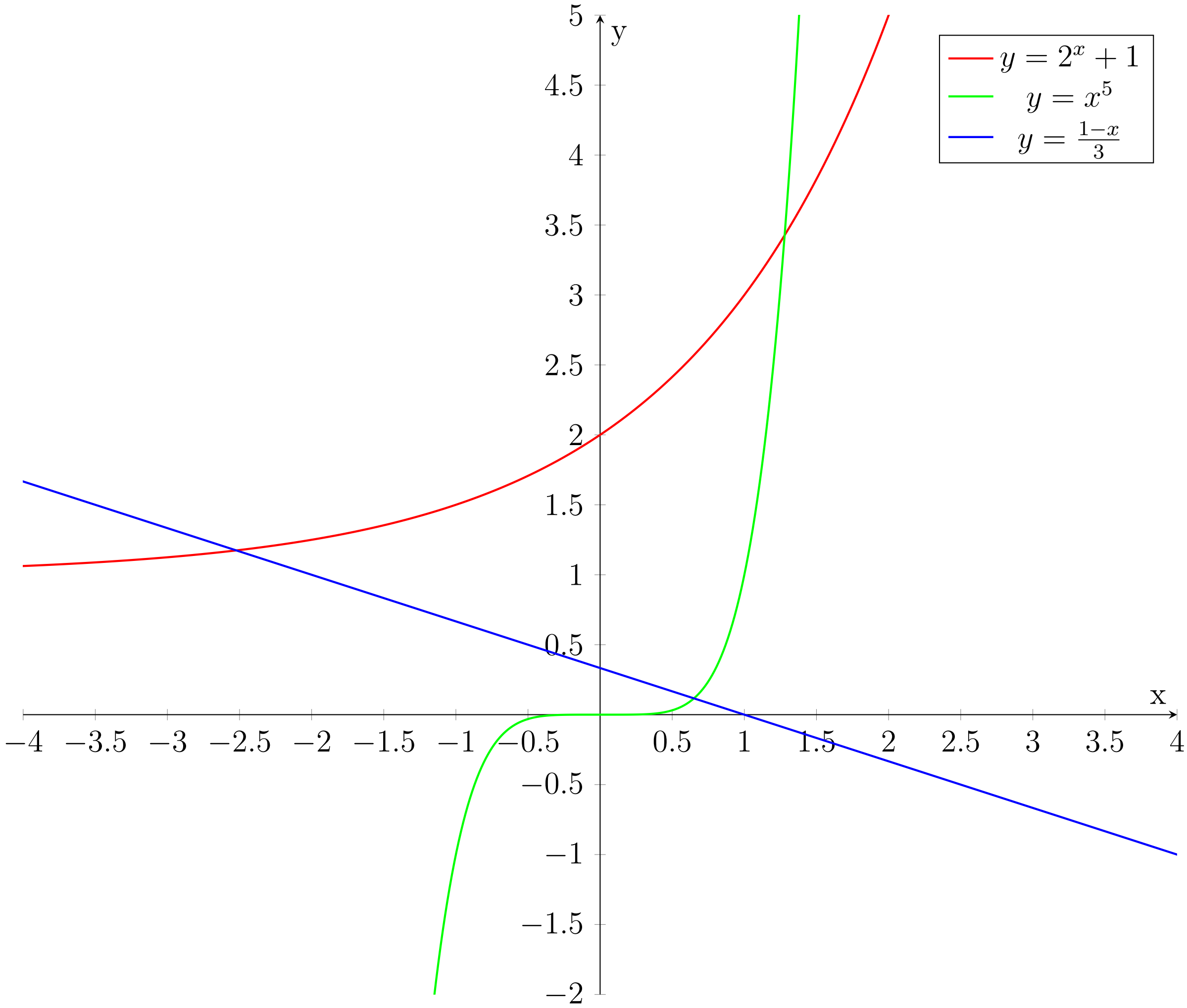

Рис. 1: Плоская фигура, ограниченная графикам заданных функций

Для корректного нахождения единственного корня уравнения $F(x) = 0$ на заданном отрезке $[a, b]$ методами деления отрезка пополам и хорд необходимо, чтобы функция F удовлетворяла на отрезке следующим свойствам[1]:

1. $F(a)F(b) < 0$;
2. F непрерывно дифференцируема на $[a, b]$;
3. F' монотонна и сохраняет знак на $[a, b]$.

Исходя из этих требований и из графика (рис. 1) разумно выбрать следующие отрезки для отыскания корней:

- $F(x) = f_1(x) - f_2(x) : [1, 2]$ (т.к. $F'(x) = 2^x \ln 2 - 5x^4$ убывает и > 0);

- $F(x) = f_1(x) - f_3(x) : [-3, -2]$ (т.к. $F'(x) = 2^x \ln 2 - \frac{1}{3}$ возрастает и < 0);
- $F(x) = f_2(x) - f_3(x) : [0.6, 1]$ (т.к. $F'(x) = 5x^4 - \frac{1}{3}$ возрастает и > 0).

Далее, для вычисления определённого интеграла методом трапеций на отрезке $[a, b]$ с n отрезками разбияния применяется следующая формула: $\int\limits_a^b f(x)\mathrm{d}x = \frac{b-a}{2n}\left(f(a) + f(b) + 2\sum\limits_{k=1}^{n-1} f(x_k)\right) + R$, где $|R| = O(\frac{1}{n^2}) \approx \varepsilon_2$. По правилу Рунге для метода трапеций: $\varepsilon_2 \approx \frac{1}{3}(I_{2n} - I_n)$, где I_n - приближённое значение (без остатка) интеграла на отрезке $[a, b]$ с n отрезками разбиения. Для определения площади потребуется вычислить 3 определённых интеграла, так что требуется увеличение точности в 3 раза. Таким образом, достаточно взять значения $\varepsilon_1 = \varepsilon_2 = 0.0001$ при $\varepsilon = 0.001$ (по условию задачи).

Результаты экспериментов

При помощи программы с точность $\varepsilon_1 = 0.0001$ найдены координаты вершин фигуры, ограниченной графиками заданных функций (таблица 1).

Кривые	x	y
1 и 2	1.2794	3.4274
1 и 3	-2.5222	1.1741
2 и 3	0.6505	0.1165

Таблица 1: Координаты точек пересечения

Далее, на рис. 2 закрашена нужная фигура и подписана её площадь, вычисленная также при помощи программы.

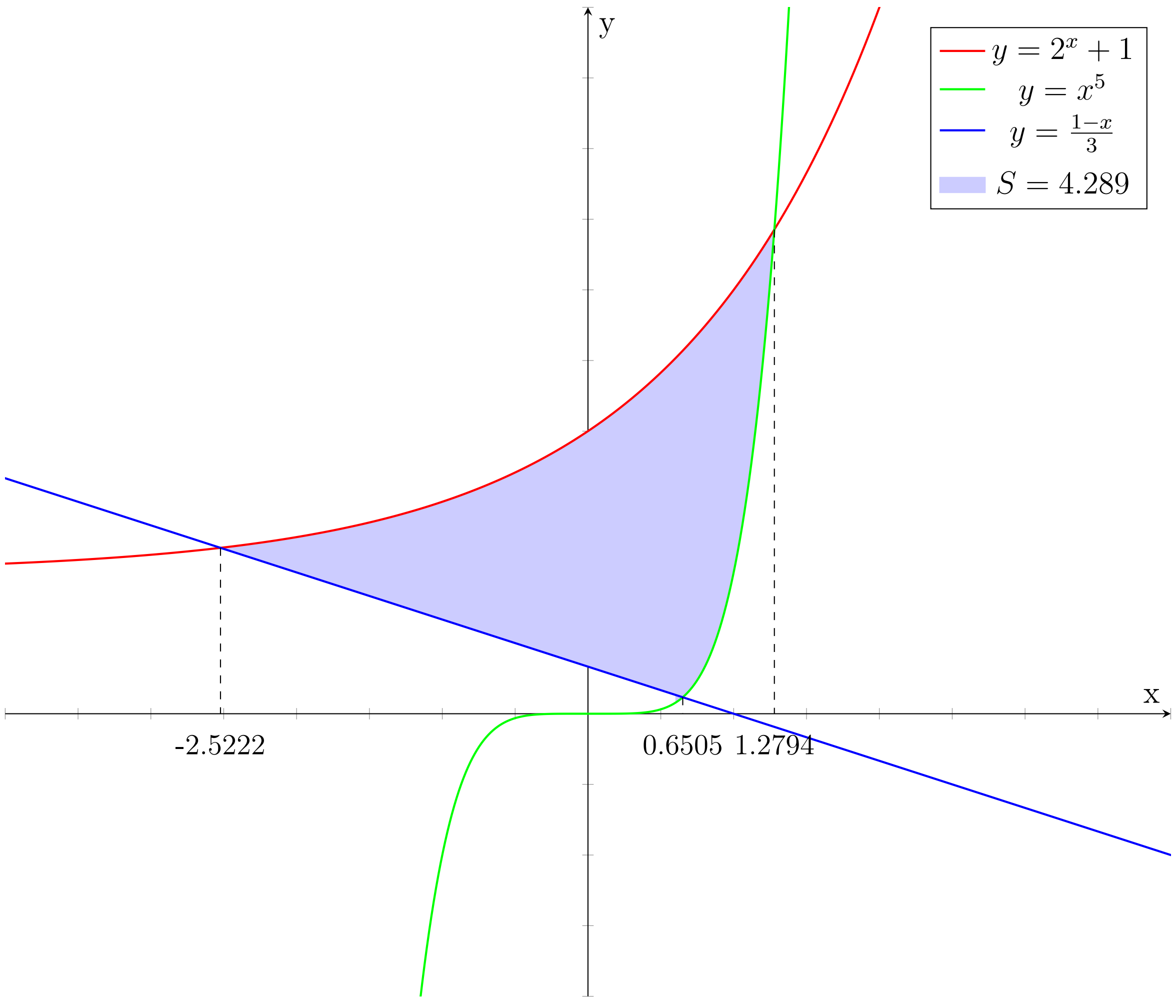

Рис. 2: Площадь плоской фигуры, ограниченной графикам заданных функций

Структура программы и спецификация функций

При работе программы используется глобальная константа `MAX_STEPS`, задающая максимальное кол-во итераций цикла поиска корня до аварийной остановки. Она нужна, т.к. при неоптимальном выборе отрезка методы поиска корня будут сходиться слишком медленно, из-за чего программа будет нерационально использовать доступные ресурсы. Также в тексте программы определяется глобальная целочисленная переменная `iter`, в которую сохраняется кол-во итераций, потребовавшееся для последнего вычисления корня.

Функции `f1`, `f2`, `f3` для вычисления математических выражений написаны на языке NASM под 32-битную архитектуру и импортируются в Си коде из объектного файла `funcs.o`.

Функция `main` анализирует параметры командной строки, при необходимости указывает пользователю корректный вариант использования или выводит полную справку. Далее, в зависимости от параметра `--mode`, возможны 3 варианта работы программы:

1. вызов функции `root` для вычисления абсцисс вершин плоской фигуры, вызов функции `integral` для вычисления определённого интеграла на интересующем отрезке под каждой функцией и вывод значения площади;

2. вызов функции `root` с пользовательскими параметрами и вывод значения корня или сообщения об ошибке;

3. вызов функции `integral` с пользовательскими параметрами и вывод значения определённого интеграла.

Полный список используемых функций:

- `double root(double (*f)(double), double (*g)(double), double a, double b, double eps1)` вычисляет корень уравнения $f(x) - g(x) = 0$ на отрезке $[a, b]$ с точностью `eps1` методом хорд или деления отрезка пополам (выбор метода осуществляется на этапе компиляции);

- `double integral(double (*f)(double), double a, double b, double eps2)` вычисляет $\int\limits_a^b f(x)\mathrm{d}x$ с точность `eps2` методом трапеций;

- `double area(int flag_inter, flag_iter, double eps)` вычисляет площадь фигуры, ограниченной кривыми, заданными функциями f_1, f_2, f_3, с точность `eps`. Флаги `flag_inter` и `flag_iter` показывают, нужно ли выводить абсциссы вершин фигуры и кол-во итераций, потребовавшихся для их нахождения, соответственно;

- `void print_usage(char *arg)` выводит справку о вариантах использования программы, параметр `arg` отвечает за название исполняемого файла;

- `void print_help(char *arg)` выводит полную справку по программе, параметр `arg` отвечает за название исполняемого файла.

Сборка программы (Make-файл)

Makefile содержит следующие основные цели:

1. `all` - сборка программы;
2. `clean` - удаление всех объектных и исполняемого файлов;
3. `arc` - создания zip-архива со всеми файлами проекта.

По заданию требуется реализовать методы хорд и деления отрезка пополам для поиска корня уравнения на заданном отрезке. Выбор метода решения определяется на этапе компиляции при помощи флага `-D`. Makefile предоставляет пользователю возможность выбора при помощи переменной `method`. При значении `SECANTS` используется метод хорд, а при значении `SEGMENTS` — метод деления отрезка пополам (он же применяется и при неправильном значении переменной `method`).

```
.PHONY all clean arc

all: main.o funcs.o main_prog

funcs.o:
	nasm -f elf32 funcs.asm -o funcs.o

main.o:
	gcc -m32 -c main.c -o main.o -D $(method)

main_prog:
	gcc -m32 -lm main.o funcs.o -o main_prog

clean:
	rm *.o main_prog

arc:
	mkdir -p ARC
	zip ‘date +%Y.%m.%d_%N‘.zip main_prog main.c Makefile funcs.asm
	mv *.zip ARC/
```

Листинг 1: Текст Makefile

Отладка программы, тестирование функций

Для тестирования и отладки основного модуля на языке Си заданы специальные режимы работы программы (только решение уравнения или только вычисление интеграла), которые задаются при помощи параметра `--mode`. В ходе тестирования полученные программой значения сравнивались с ответами Wolfram Mathematica.

Отладка функций, написанных на языке Ассемблера производилась при помощи инструментов SASM. В частности, велось наблюдение за значениями регистров сопроцессора x87.

Программа на Си и на Ассемблере

Полный код программы, включащий в себя модули на языках Си и Ассембелер, а также Makefile, приложен к отчёту и находится в том же архиве. Си код содержится в файле `main.c`, реализация математических функций на языке Ассемблера — в файле `funcs.asm`.

Анализ допущенных ошибок

При написании модуля на языке Ассемблера сначала были допущены ошибки, связанные с вычислением 2^x, а также с тем, что возвращаемое значение сохранялось в регистре `eax`, а не в `st0`.

Кроме того, изначально функция `root` не учитывала характер выпуклости функции $\varphi = f - g$, из-за чего могла уйти в бесконечный цикл.

Список литературы

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