

## ЗАДАНИЕ № 6 ПО ПРАКТИКУМУ

для студентов 1 потока 1 курса ф-та ВМК МГУ  
в 2015/2016 учебном году, весенний семестр

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

**Языки программирования: Си, ассемблер NASM.**

### ПОСТАНОВКА ЗАДАЧИ

С заданной точностью  $\varepsilon$  вычислить площадь плоской фигуры, ограниченной тремя кривыми, уравнения которых  $y = f_1(x)$ ,  $y = f_2(x)$  и  $y = f_3(x)$  либо заранее определены вариантом задания либо задаются в текстовом виде на этапе сборки программы. Во втором случае необходимо разработать две программы: основную – для вычисления площади и вспомогательную – для построения исполняемого кода, вычисляющего значения функций.

При решении задачи необходимо следующее.

- С некоторой точностью  $\varepsilon_1$  вычислить абсциссы точек пересечения кривых, используя предусмотренный вариантом задания метод приближенного решения уравнения  $F(x)=0$ . Во вариантах задания, где уравнения кривых фиксированы, отрезки, где программа будет искать точки пересечения и где применим используемый метод, следует определить вручную. Во вариантах задания, где уравнения задаются в текстовом виде, отрезки для поиска пересечения задаются вместе с уравнениями.
- Представить площадь заданной фигуры как алгебраическую сумму определенных интегралов и вычислить эти интегралы с некоторой точностью  $\varepsilon_2$  по квадратурной формуле, предусмотренной вариантом задания.

Величины  $\varepsilon_1$  и  $\varepsilon_2$  подобрать вручную так, чтобы гарантировалось вычисление площади фигуры с точностью  $\varepsilon$ .

### ВАРИАНТЫ ЗАДАНИЯ

**Замечание:** варианты, помеченные вертикальной чертой слева, имеют повышенную сложность.

#### I. Уравнения кривых

Варианты 1-10, обычной сложности. Точность вычислений  $\varepsilon = 0.001$ .

$y = f_i(x)$ :

1) $f_1 = 2^x + 1$ ,	$f_2 = x^5$ ,	$f_3 = (1 - x)/3$
2) $f_1 = 3(0.5/(x + 1) + 1)$ ,	$f_2 = 2.5x - 9.5$ ,	$f_3 = 5/x \ (x > 0)$
3) $f_1 = \exp(-x) + 3$ ,	$f_2 = 2x - 2$ ,	$f_3 = 1/x$
4) $f_1 = \exp(x) + 2$ ,	$f_2 = -1/x$ ,	$f_3 = -2(x + 1)/3$
5) $f_1 = 0.35x^2 - 0.95x + 2.7$ ,	$f_2 = 3x + 1$ ,	$f_3 = 1/(x + 2)$
6) $f_1 = 0.6x + 3$ ,	$f_2 = (x - 2)^3 - 1$ ,	$f_3 = 3/x$
7) $f_1 = \ln(x)$ ,	$f_2 = -2x + 14$ ,	$f_3 = 1/(2 - x) + 6$
8) $f_1 = \exp(x) + 2$ ,	$f_2 = -2x + 8$ ,	$f_3 = -5/x$
9) $f_1 = 3/((x - 1)^2 + 1)$ ,	$f_2 = \text{sqrt}(x + 0.5)$ ,	$f_3 = \exp(-x)$
10) $f_1 = 1 + 4/(x^2 + 1)$ ,	$f_2 = x^3$ ,	$f_3 = 2^{-x}$

Вариант 11, повышенной сложности.

Плоская фигура образуется графиками полиномов, отрезок, на котором графики этих полиномов пересекаются и образуют плоскую фигуру, задаются в текстовом файле. Точности вычислений  $\varepsilon_1$  и  $\varepsilon_2$  в этом случае не подбираются самостоятельно, а задаются в опциях основной программы.

Полиномы имеют вид  $y = \sum a_i x^{b_i}$ , где коэффициенты  $a_i$  и  $b_i$  – вещественные числа. Коэффициенты полиномов, границы отрезка и точность вычислений задаются в текстовом файле, формат файла определяется самостоятельно и исчерпывающе описывается в тексте отчета.

Вспомогательная программа считывает содержимое текстового файла и по уравнениям полиномов строит ассемблерный листинг функций f1, f2, f3 а также их производных (если таковое необходимо). В теле функций командами x87 осуществляется вычисление значения  $y$  для соответствующего значения  $x$ .

Построенный листинг транслируется ассемблером `nas` в перемещаемый объектный код, который затем используется при компоновке основной программы.

## II. Методы приближенного решения уравнений

1. Метод деления отрезка пополам;
2. метод хорд (секущих);
3. метод касательных (Ньютона);
4. комбинированный метод (хорд и касательных).
5. Реализуется два различных метода решения уравнений. Конкретный метод выбирается во время препроцессорирования с использованием условной компиляции и директив препроцессора (`#ifdef`, `#ifndef`, ...).

## III. Квадратурные формулы

1. формула прямоугольников,
2. формула трапеций,
3. формула Симпсона (парабол).

Таким образом, вариант определяется следующими параметрами: (1) способом задания кривых и их набором для вариантов 1-10, (2) методом приближенного решения уравнений, (3) квадратурными формулами.

## ТРЕБОВАНИЯ К ПРОГРАММЕ

1. Функции, вычисляющие значения  $f_1$ ,  $f_2$ ,  $f_3$ , и их производных (в случае необходимости производных) реализуются на языке ассемблера с соглашением вызова `cdecl`. Все остальные функции программы реализуются на языке Си.
2. Основная программа должна поддерживать опции командной строки, при задании которых:
  - а) печатаются абсциссы точек пересечения кривых,
  - б) печатается число итераций, потребовавшихся на приближенное решение уравнений при поиске точек пересечения.
3. Программа должна поддерживать ключ командной строки `-help`, выводящий на печать все допустимые ключи командной строки.
4. Вычисление с точностью  $\varepsilon_1$  корня  $x$  уравнения  $f(x) = g(x)$  на отрезке  $[a, b]$  должно быть реализовано в отдельной Си-функции `root(f, g, a, b, eps1, x)`. Если используется метод касательных или комбинированный метод, то у `root` должно быть еще два параметра функционального типа, позволяющие вызывать производные функций  $f$  и  $g$ .
5. Вычисление с точностью  $\varepsilon_2$  величины определенного интеграла от функции  $f(x)$  на отрезке  $[a, b]$  должно быть реализовано в отдельной Си-функции `integral(f, a, b, eps2)`.
6. Си-функции `root` и `integral` должны быть предварительно протестированы. Основная программа должна предоставлять возможности тестирования, активируемые опцией командной строки.

Фактические параметры вызова тестируемых Си-функций задаются в командной строке; параметры функционального типа задаются по номеру соответствующей функции.

7. Сборка программы должна осуществляться при помощи утилиты `make`. Соответствующий файл должен явно или неявно описывать зависимости между всеми целями сборки. Должны быть определены цели *all* и *clean*, первая из которых полностью собирает программу, а вторая — удаляет все промежуточные файлы (в частности, объектные модули). Сдаваемый архив должен включать в себя `Makefile`.
8. Программа сдаётся в виде `.zip`-архива, содержащего в себе все необходимые файлы с исходным кодом, а также отчёт в формате PDF.
9. Программа должна быть снабжена поясняющими комментариями в объёме, достаточном для её понимания. Все глобальные и статические переменные должны быть документированы в комментариях.
10. Выбор конкретного метода решения уравнений должен управляться определёнными символами на этапе препроцессирования и передаваться через ключ `-D`.

## ЛИТЕРАТУРА

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

## МЕТОДИЧЕСКИЕ УКАЗАНИЯ

1. Методические указания о численных методах и их реализации в программе приведены в методическом пособии «Задания практикума на ЭВМ» Трифонов Н.П., Пильщиков В.Н., задание 1.
2. Необходимо помнить о том, что внешние имена на платформе Windows снабжаются ведущим подчёркиванием. На этой платформе функцию `f1` следует в Ассемблерном коде называть `_f1`. Си-код при этом не меняется. К сдаче должен быть подготовлен вариант программы, предназначенный для запуска в UNIX-окружении.
3. Аналитическую работу с кривыми упрощают онлайн-сервисы, например, <http://fooplot.com/>