



Язык программирования Си

Бабарыкин Д. С. Кафедра ИУ7

19.09.18

Методические указания к лабораторной работе № 3

1. Введение в итерационные задачи

Итерация – единократное выполнение тела цикла. Работа цикла в информатике заключается в многократном выполнении заданного кода (тела цикла). Таким образом, цикл состоит из множества итераций – проходов по телу цикла.

Итерационные задачи – тип задач в программировании, решение которых достигается за счет многократного циклического повторения одного и того же участка алгоритма, причем с каждой новой итерацией ответ становится всё более точным. Важной особенностью такого типа задач является поиск компромисса между точностью полученного результата и временем выполнения программы (количеством итераций). Как правило, ответ ищется с заданной погрешностью – максимально допустимой разницей ответа между двумя последовательными итерациями. Пускай

S_i – результат, полученный на i -ой итерации

S_{i-1} – результат, полученный на $(i-1)$ -ой итерации (на предыдущей)

E – погрешность

тогда итерационный процесс можно остановить при выполнении условия:

$$|S_i - S_{i-1}| < E$$

При наступлении этого момента объявляется, что задача была решена с результатом S_i при погрешности итерационного процесса E .

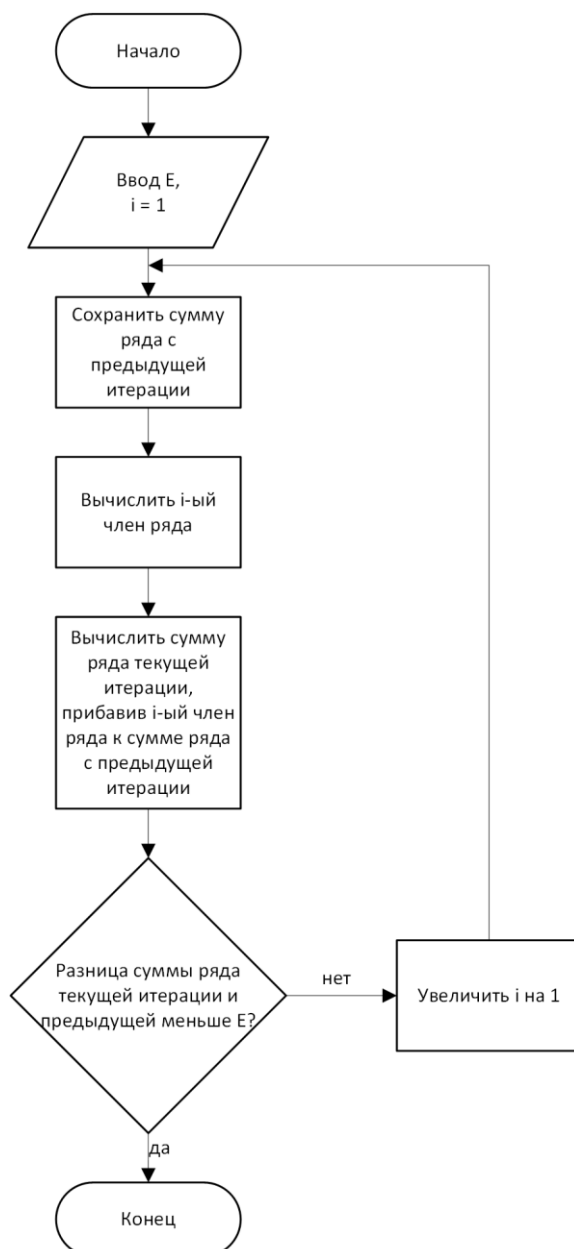
2. Вычисление суммы бесконечного ряда

Одной из задач, относящихся к типу итерационных, является вычисление суммы сходящегося бесконечного ряда. Примеры сходящихся бесконечных рядов:

$$\sum_{n=1}^{\infty} \frac{1}{n} = \frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots$$

$$\sum_{n=3}^{\infty} \frac{5}{3n^2 + 1} = \frac{5}{3 \times 3^2 + 1} + \frac{5}{3 \times 4^2 + 1} + \frac{5}{3 \times 5^2 + 1} + \frac{5}{3 \times 6^2 + 1} + \dots$$

Пусть необходимо вычислить сумму какого-то сходящегося бесконечного ряда с точностью E . Тогда алгоритм будет выглядеть следующим образом:



3. Вычисление интеграла

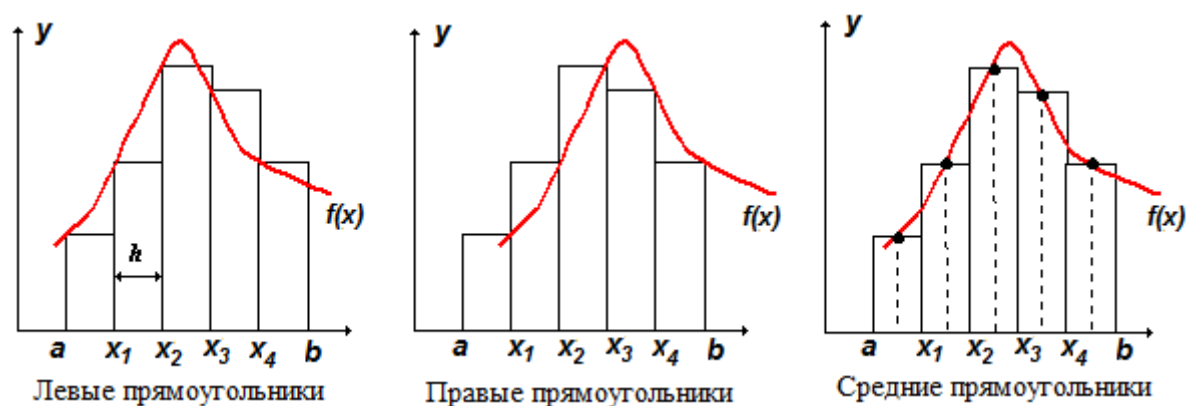
3.1. Описание задачи

Задача вычисления определенного интеграла также может быть решена с помощью итерационного метода благодаря следующему свойству:

Определенный интеграл от непрерывной функции численно равен площади фигуры под графиком данной функции на заданном отрезке.

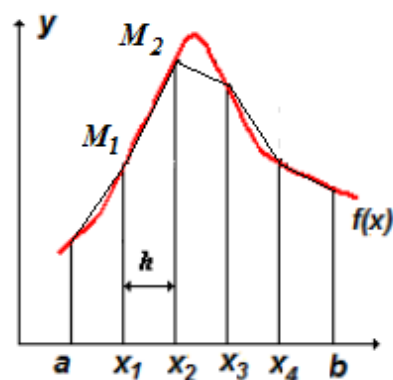
Таким образом, задача вычисления интеграла сводится к итерационной задаче нахождения площади фигуры под графиком. Но рассчитать площадь получаемой фигуры аналитически довольно проблематично за счет криволинейности её поверхности. Поэтому будем аппроксимировать её фигурами, чью площадь рассчитать довольно просто: прямоугольниками и трапециями. Если их основания будут небольшими, то можно утверждать, что сумма их площадей приближенно равна площади начальной фигуры. Рассмотрим четыре самых простых метода: левых прямоугольников, правых прямоугольников, средних прямоугольников и трапеций.

Графически методы прямоугольников выглядят следующим образом:



Как видно из рисунка, все эти методы отличаются лишь тем, в каких местах прямоугольники касаются графика функции, а значит это влияет на то, в каких точках необходимо вычислять значение функции для расчета площади очередного прямоугольника.

Метод трапеций аналогичен методам с прямоугольниками с единственной разницей, что фигурами с вычисляемой площадью в нём служат трапеции:



Однако однократного приближенного расчета площади фигуры под графиком путем аппроксимации её на заданное количество прямоугольников или трапеций, как правило, недостаточно для выявления точного значения интеграла. И здесь нам на помощь приходит метод итераций. Для расчета интеграла он заключается в следующем:

1. Задается начальное количество участков разбиения N (оно равно количеству фигур, которыми аппроксимируется первоначальная фигура)
2. Рассчитывается приближенная площадь фигуры одним из способов (прямоугольников или трапеций) с заданным количеством участков разбиений N
3. Количество участков разбиений N удваивается и расчет площади повторяется, но уже для нового N
4. Полученное значение площади сравнивается с предыдущим и если модуль их разницы меньше заданного значения погрешности E , то процесс останавливается, иначе возврат к пункту 3.

3.2. Формулы расчётов для методов прямоугольников и трапеций

Условные обозначения:

S – значение интеграла и суммарная площадь фигуры под графиком

S_i – площадь i -ого прямоугольника или трапеции

N – количество участков разбиения, равное количеству аппроксимирующих фигур

a, b – пределы интегрирования, являющиеся правой и левой границей общей фигуры

h – шаг (приращение функции), равное основанию прямоугольников и трапеций, вычисляется по формуле: $h = \frac{b-a}{N}$

x_i – точка на оси X , в которой рассчитывается высота i -ого прямоугольника

$$x_0 = a, x_1 = a + h$$

Метод левых прямоугольников:

$$S = \sum_{i=1}^N S_i = \sum_{i=1}^N h \times f(x_{i-1})$$

Метод правых прямоугольников:

$$S = \sum_{i=1}^N S_i = \sum_{i=1}^N h \times f(x_i)$$

Метод средних прямоугольников:

$$S = \sum_{i=1}^N S_i = \sum_{i=1}^N h \times f\left(\frac{x_{i-1} + x_i}{2}\right)$$

Метод трапеций:

$$S = \sum_{i=1}^N S_i = \sum_{i=1}^N h \times \frac{f(x_{i-1}) + f(x_i)}{2}$$

4. Задание на лабораторную работу

4.1. Расчёт суммы бесконечного ряда

Написать программу, вычисляющую сумму сходящегося бесконечного ряда S с точностью до E , которое вводит пользователь. Ряд:

$$S = \sum_{n=1}^{\infty} \frac{1}{n!}$$

Так как ряд имеет довольно характерный вид, то вычисление следующего члена ряда на основе предыдущего является более предпочтительным путем, нежели полное вычисление факториала на каждой итерации.

4.2. Расчёт интеграла

Написать программу, вычисляющую значение определенного интеграла S с точностью до E от функции натурального логарифма:

$$S = \int_a^b \ln x \, dx$$

Пределы интегрирования a и b , а также максимально допустимая погрешность E вводятся пользователем.

Расчёт должен осуществляться одним из заданных методов по вариантам:

- 1) Метод левых прямоугольников
- 2) Метод правых прямоугольников
- 3) Метод средних прямоугольников
- 4) Метод трапеций

Вариант рассчитывается по формуле:

$$\text{Вариант} = (\text{Номер в списке группы}) \bmod 4$$

Студентам, которые хотят проявить себя, рекомендуется самостоятельно рассмотреть чуть более сложный, чем представленные выше, метод расчёта интегралов – [метод парабол \(Симпсона\)](#) и реализовать в программе вместо предложенных именно его.