БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

Лабораторная работа по теме:   
**Интегрирование функции на отрезке**

Выполнил:  
 Ткачук П.А.  
 2 курс 1 группа

Преподаватель:  
 Будник А.М.

Дата:

30.05.2017

*Минск*

# Постановка задачи

Для функции необходимо:

1. Используя формулу левых прямоугольников, вычислить по правилу Рунге интеграл с точностью ε на промежутке [a, b]
2. Вычислить шаг h в формулах средних прямоугольников и трапеций, достаточный для получения результата с точностью ε на [a, b].
3. Найти количество узлов n квадратурной формулы НАСТ, достаточное для достижения точность ε на [a, b].

# Входные данные

ε = 0.0001, a = 0, b = 1

# Расчетные формулы

1. **Общий алгоритм интегрирования по формуле левых прямоугольников с использованием метода Рунге**

Формула для интеграла с шагом h: , где

Формула Рунге с помощью которой оценивается точность вычисленного интеграла:

, где , – интеграл вычисленный по вышеприведённой формуле с шагом .

Делаем итерирование по i от 1 до n. И если на i-м шаге получили необходимую оценку точности, то искомым интегралом является .

1. **Общий алгоритм оценки шага h для достижения точности ε**

Чтобы найти шаг необходимый для достижения точности ε в формулах средних прямоугольников и трапеции воспользуемся оценками их погрешностей:

*- для формулы средних прямоугольников*

*- для формулы трапеций*

Из них получаем необходимые оценки h.

1. **Общий алгоритм интегрирования по квадратурной формуле Гаусса**

В своём конечном виде квадратурное правило Гаусса интегрирования функции на отрезке [-1 , 1] представляет из себя взаимно однозначное соответствие между количеством используемых на отрезке узлов и значениями этих узлов с коэффициентами для квадратурной формулы. Имеются таблицы узлов и коэффициентов квадратурной формулы Гаусса, поэтому достаточно лишь для желаемого количества узлов посмотреть в таблице требуемые значения и подставить их в квадратурную формул:

где – коэффициенты квадратурной формулы Гаусса; – узлы квадратурной формулы.

Для произвольного промежутка формула принимает вид:

Погрешность в таком случае оценивается по формуле:

# Программная реализация

**import** math  
**import** numpy **as** np  
**import** scipy.special **as** sp  
  
f **= lambda** x**:** math.log(1. **+** 2. **\*** x) **/** x **if** x **!=** 0 **else** 2.  
a **=** 0  
b **=** 1  
eps **=** 0.00001  
M **=** 16. **/** 3.  
gauss **=** [[2.],  
 [1., 1.],  
 [5. **/** 9., 8. **/** 9., 5. **/** 9.],  
 [0.3478548451, 0.65214515490, 0.65214515490, 0.3478548451],  
 [0.2369268851, 0.4786286705, 0.5688888899, 0.4786286705, 0.2369268851],  
 [0.1713244924, 0.3607615730, 0.4679139346, 0.4679139346, 0.3607615730, 0.1713244924]]  
  
roots **=** [[0.],  
 [**-**0.5773502692, 0.5773502692],  
 [**-**0.7745966692, 0., 0.7745966692],  
 [**-**0.8611363116, **-**0.3399810436, 0.3399810436, 0.8611363116],  
 [**-**0.9061798459, **-**0.5384693101, 0., 0.5384693101, 0.9061798459],  
 [**-**0.9324695142, **-**0.6612093865, **-**0.2386191861, 0.2386191861, 0.6612093865, 0.9324695142]]  
  
  
**def runge**(*f*, *a*, *b*, *eps*, *integral*, *m*)**:** h, h\_next **=** 1., 1.  
 is\_enough **= False** I, I\_next **=** 0, 0  
 **while not** is\_enough**:** h\_next **=** h **/** 2.  
 I **=** *integral*(*f*, *a*, *b*, h)  
 I\_next **=** *integral*(*f*, *a*, *b*, h\_next)  
 is\_enough **=** abs((I\_next **-** I) **\*** (h\_next **\*\*** *m*) **/** (h **\*\*** *m* **-** h\_next **\*\*** *m*)) **<** *eps* h **=** h\_next  
 **return** I\_next  
  
  
**def trapezium\_h**(*M*, *a*, *b*, *eps*)**:  
 return** math.sqrt(12 **\*** *eps* **/** (*M* **\*** (*b* **-** *a*)))  
  
  
**def middle\_rect\_h**(*M*, *a*, *b*, *eps*)**:  
 return** math.sqrt(24 **\*** *eps* **/** (*M* **\*** (*b* **-** *a*)))  
  
  
**def left\_rect\_with\_runge**(*f*, *a*, *b*, *eps*)**:  
 def left\_rect**(*f*, *a*, *b*, *h*)**:** n **=** (int((*b* **-** *a*) **/** *h*))  
 **return** *h* **\*** sum([*f*(*a* **+** *h* **\*** i) **for** i **in** range(n)])  
  
 **return** runge(*f*, *a*, *b*, *eps*, left\_rect, 1)  
  
  
**def gauss\_n**(*I*, *f*, *a*, *b*, *eps*)**:** transform **= lambda** *x***:** (*b* **-** *a*) **\*** x **/** 2. **+** (*b* **+** *a*) **/** 2.  
 enough **= False** n **=** 0  
 **while not** enough**:** I\_gauss **=** (*b* **-** *a*) **\*** sum([gauss[n][i] **\*** *f*(transform(x\_i)) **for** i, x\_i **in** enumerate(roots[n])]) **/** 2.  
 **if** (abs(I\_gauss **-** *I*) **<=** *eps*)**:** enough **= True** n **+=** 1  
 **return** n  
  
  
I **=** left\_rect\_with\_runge(f, a, b, eps)  
print(**'Integral with accuracy eps = %.5f calculated using left rectangle method\n--- I=%.10f' %**(eps, I))  
print(**'The step in middle rectangle method to achieve accuracy eps = %.5f\n--- h=%.10f' %** (eps, middle\_rect\_h(M, a, b, eps)))  
print(**'The step in trapezium method to achieve accuracy eps = %.5f\n--- h=%.10f' %** (eps, trapezium\_h(M, a, b, eps)))  
print(**'Amount of nodes to achieve highest algebraic accuracy power\n--- n=%d' %**gauss\_n(I, f, a, b, eps))

# Результат работы программы

Integral with accuracy eps = 0.00001 calculated using left rectangle method

--- I=1.4367532440

The step in middle rectangle method to achieve accuracy eps = 0.00001

--- h=0.0067082039

The step in trapezium method to achieve accuracy eps = 0.00001

--- h=0.0047434165

Amount of nodes to achieve highest algebraic accuracy power

--- n=5

# Вывод

Вычисленный программой интеграл отличается от истинного значения на 6.87707289204198e-06 и из этого следует, что необходимая точность обеспечивается.

Сравнивая шаги интегрирования, вычисленные программой для метода средних прямоугольников и метода трапеций, ожидаемо получили что для метода прямоугольников шаг больше. Это следует из того, что функция R(h) погрешности первого метода выдает значение меньшее чем погрешность второго метода.

Также получили что для метода Гаусса необходимо взять 5 узлов. Сравнивая это количество с количеством необходимым для интегральных формул прямоугольников и трапеций видим, что Гаусс дал значительный выигрыш в количестве узлов (для метода прямоугольников n150, для метода трапеций n210).