Реализация численного интегрирования с помощью квадратурной формулы Гаусса-Кристофелля. Квадратурная форма Гаусса (1.3.10L)

1. Суть метода и алгоритм решения:

Дана непрерывная на некотором промежутке $[a; b]$ функция $f(x)$.

Пусть существует какая-то весовая функция $p(x) > 0$, которая непрерывна на интервале $(a; b)$.

Рассматривается задача поиска определенного интеграла вида:

$$I = \int_a^b f(x)p(x)dx$$

Данный метод необходим, когда выразить интеграл через элементарные функции в общем случае либо не удается, либо вызывает слишком много проблем. Поэтому обычно $f(x)$ заменяют на некоторую аппроксимирующую функцию $\phi(x) = f(x)$. Она подбирается таким образом, чтобы интеграл от нее легко считался в элементарных функциях. Стандартный пример $\phi(x) -$ некоторый обобщенный интерполяционный многочлен. При этом $f(x)$ заменяется линейным выражением со значениями в узлах в качестве некоторых коэффициентов:

$$f(x) = \sum_{k=1}^{n} f(x_k)\phi_k(x) + r(x),$$

где $r(x)$ – некоторый остаточный член аппроксимации, $n -$ количество узлов аппроксимации.

Подставив только что выраженную функцию $f(x)$ в интеграл получаем следующее:

$$I = \sum_{k=1}^{n} f(x_k)A_k + R,$$

где $A_k = \int_a^b \phi(x)p(x)dx$, $R = \int_a^b r(x)p(x)dx$.

В нашем случае, R=0

Целью данной работы является реализация одного из частного вида данного метода: квадратурная формула Гаусса, определенная на $[-1; 1]$.

Для некоторых случаях n вычислены значения A_k и x_k, по котором будем вычислять интегралы.

n=2	$x_{0,1} = \pm .57735027$ $A_{0,1} = 1$
n=3	$x_{0,2} = \pm .7745966692$ $A_{0,2} = 5/9$ $x_1 = 0$ $A_1 = 8/9$
n=4	$x_{0,3} = \pm .8611363116$ $A_{0,3} = 0.34785485$ $x_{1,2} = \pm .3399810436$ $A_{1,2} = 0.6521451549$
n=5	$x_{0,4} = \pm .9061798459$ $A_{0,4} = 0.2869268851$ $x_{1,3} = \pm .5384693101$ $A_{1,3} = 0.4786286705$ $x_2 = 0$ $A_2 = 0.568888899$

$p(x) = 1$, $[a; b] = [-1; 1]$

2. Программная реализация:

Входные данные: функция $f(x)$ и количество узлов n.

Выходные данные: найденное
численное значение интеграла $\int_{-1}^{1} f(x)p(x)dx$

Данный метод реализован с помощью Wolfram Mathematica:

```
n
dt = {
  {{.57735027, 1}, {-0.57735027, 1}},
  {{.7745966692, 5 / 9}, {0, 8 / 9}, {-.7745966692, 5 / 9}},
  {{.8611363116, .34785485}, {.3399810436, -.6521451549},
   {-.3399810436, -.6521451549}, {-.8611363116, .34785485}},
  {{.9061798459, 0.2869268851}, {-.9061798459, 0.2869268851},
   {.5384693101, 0.4786286705}, {-.5384693101, 0.4786286705},
   {0, 0.5688888899}}
 }
Total@Table[i[[2]] * f /. d → i[[1]], {i, dt[[n - 1]]}]
```

3. Сравнение и анализ результатов.

Рассмотрим несколько различных функций. Сравним точность вычислений со встроенными функциями *Wolfram Mathematica* и рассмотрим, как влияет значение n на полученный результат.

а) $f(x) = 5x\sin(x)$

При $n = 2$:

Численный результат вычисления собственной функции: 3.15121

Численный результат вычисления встроенной функции: 3.01169

При $n = 3$:

Численный результат вычисления собственной функции: 3.00986

Численный результат вычисления встроенной функции: 3.01169

При $n = 5$:

Численный результат вычисления собственной функции: 3.36834

Численный результат вычисления встроенной функции: 3.01169

Б) $f(x) = e^{\sin(x)}$

При $n = 2$:

Численный результат вычисления собственной функции: 2.30537

Численный результат вычисления встроенной функции: 2.28319

При $n = 3$:

Численный результат вычисления собственной функции: 2.28304

Численный результат вычисления встроенной функции: 2.28319

При $n = 5$:

Численный результат вычисления собственной функции: 3.36834

Численный результат вычисления встроенной функции: 3.01169

в) $f(x) = -x^2 e^{\sin(x)} + 5$

При $n = 2$:

Численный результат вычисления собственной функции: 9.23154

Численный результат вычисления встроенной функции: 9.16912

При $n = 3$:

Численный результат вычисления собственной функции: 9.16351

Численный результат вычисления встроенной функции: 9.16912

При $n = 5$:

Численный результат вычисления собственной функции: 9.56023

Численный результат вычисления встроенной функции: 9.16912

По данным результатам видно, что алгоритм работает достаточно точно при относительно малых значениях n. Однако существует некая погрешность