|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| Федеральное государственное бюджетное  образовательное учреждение высшего образования «Новосибирский государственный технический университет» | | | |
|  | | | |
| Кафедра теоретической и прикладной информатики | | | |
|  | | | |
| Практическое задание № 4 | | | |
| по дисциплине «Программирование вычислений» | | | |
| **Численное интегрирование** | | | |
|  | | | |
|  | Группа | ПМ-81 |
| Бригада | 4 |
|  | Басов Денис |
|  | Юрганов Егор |
| Преподаватель | рояк михаил эммануилович |
|  |  |
|  |  |
|  | | | |
| Новосибирск  2020 | | | |

1. **Цель работы**

Изучение методов численного интегрирования, оценки порядка точности, оценки погрешности по правилу Рунге, уточнения значений по Ричардсону

1. **Ход работы**

**Вариант:** метод прямоугольников, метод Симпсона, метод Гаусса-4.

1. Разработать программу в соответствии с требованиями.
2. Для каждого из методов провести верификацию программы с помощью полиномов соответствующих степеней. Заполнить соответствующую таблицу.

Для метода прямоугольников: порядок аппроксимации k - 2, порядок точности m - 1.

Для метода Симпсона: порядок аппроксимации k - 4, порядок точности m - 3.

Для метода Гаусса-4: порядок аппроксимации k - 8, порядок точности m - 7.

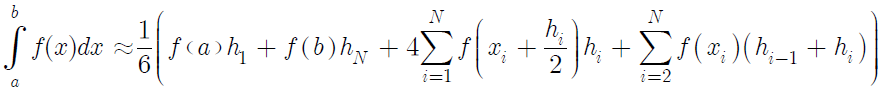
1. Для таблиц из пункта 2 провести округления чисел в соответствии с правилами В.М. Брадиса.
2. Провести исследование порядка аппроксимации изучаемого метода интегрирования для нескольких функций(полином степени k+3, осциллирующая функция). Заполнить соответствующую таблицу. Количество строк в таблице определить следующим образом: уменьшать шаг в двое до тех пор, пока погрешность уменьшается, затем сделать еще 3 дробления.
3. Исследовать влияние длины слова при реализации численного интегрирования. Провести исследования с двойной точностью.
4. **Анализ задачи**

Основные формулы, использованные при выполнении лабораторной работы:

1. Усложнённая квадратурная формула прямоугольников:

https://sun9-64.userapi.com/IiAshZEMzARFEsD0Eh1412E-mYKxqJWGXhxOnA/DJCI5uVqoiY.jpg

1. Усложнённая квадратурная формула Симпсона:



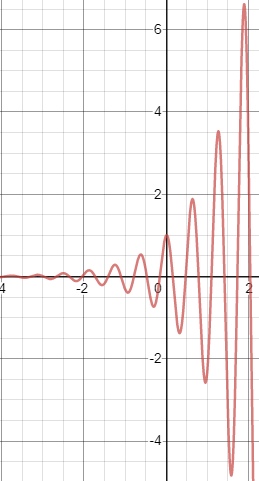
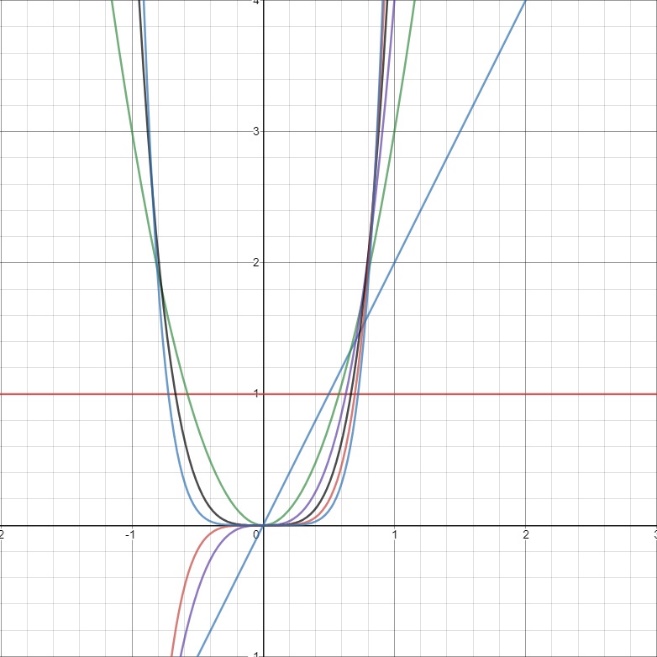
1. Усложнённая квадратурная формула Гаусса:

https://sun4-15.userapi.com/WNJaLH09uP6JRp1G7qHqBGTG0t8GEa7Enb5AiA/Qf4sNLj3OKc.jpg 

https://sun9-54.userapi.com/4mQT9zap8YYzwQiChzDD9VksozN-mxAcr7_y5Q/0mwzqZPNnB8.jpghttps://sun9-44.userapi.com/DelezlylJHiBrSwrdwWuvU2fVI711gegMdQdkw/-GvjNDuEzNM.jpg

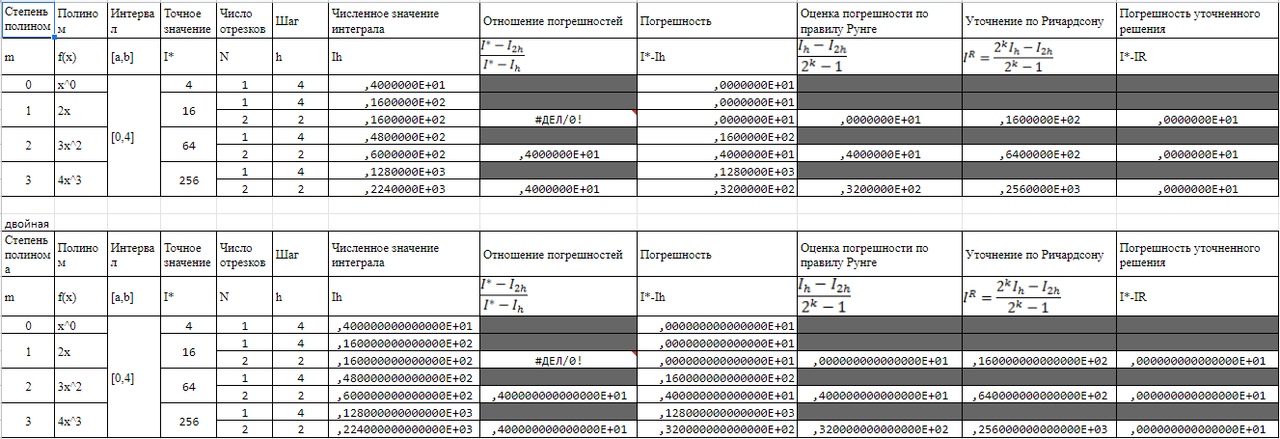
https://sun9-31.userapi.com/48wHAQSo4PWdq6AWX1LMP_4aXrWviMb0rsSjhw/V7wmsKfb63Y.jpg

Сильно осциллирующая функция

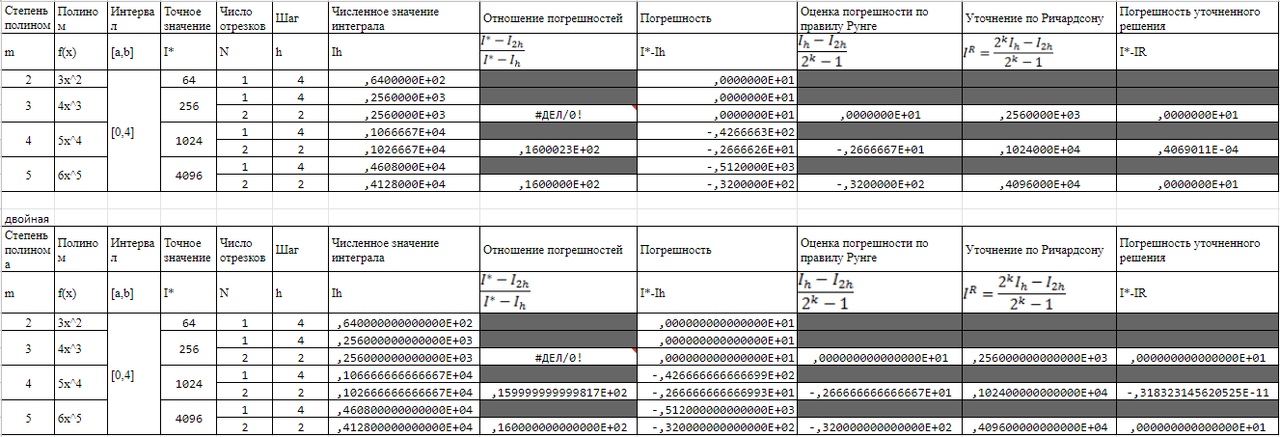


1. **Верификация программы с помощью полиномов**

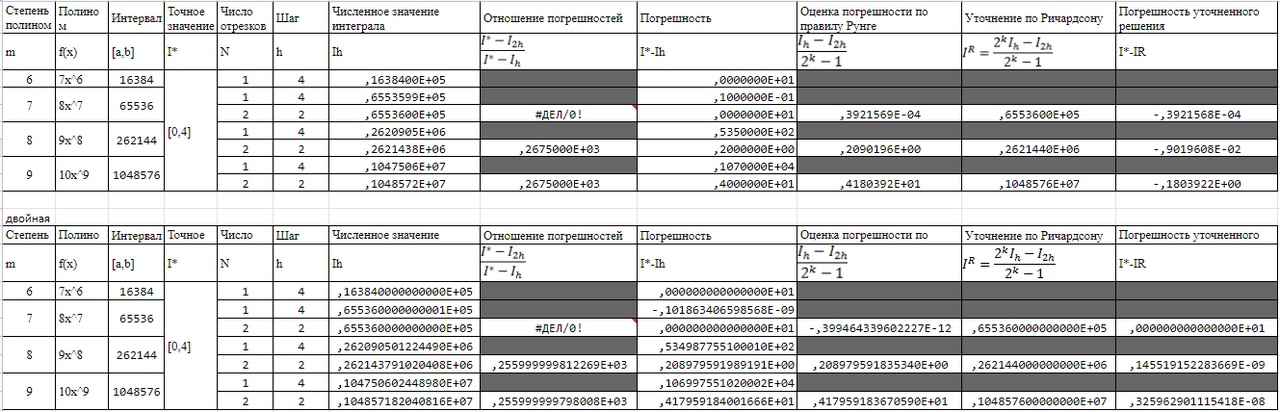
Метод прямоугольника (порядок аппроксимации k=2)

****

Метод Симпсона (порядок аппроксимации k=4)

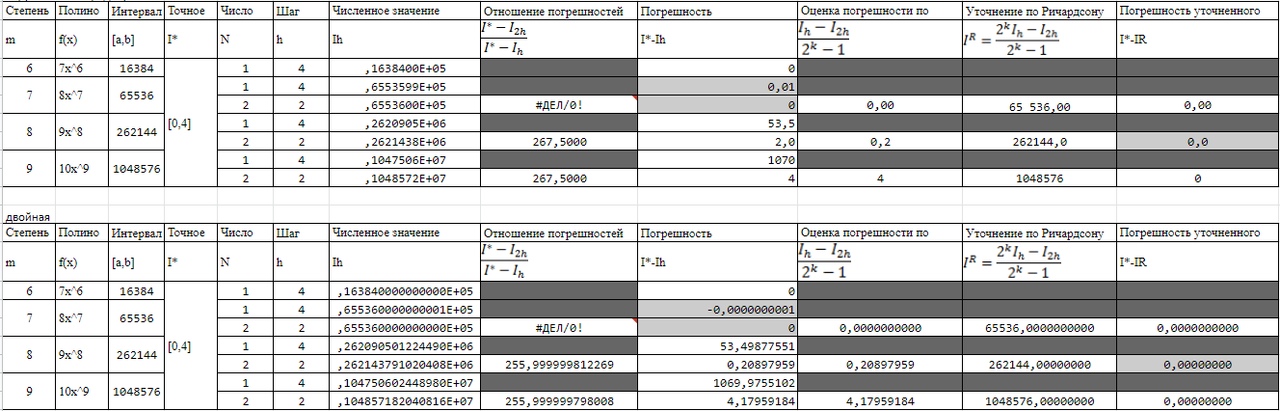
****

Метод Гаусс-4 (порядок аппроксимации k=8)

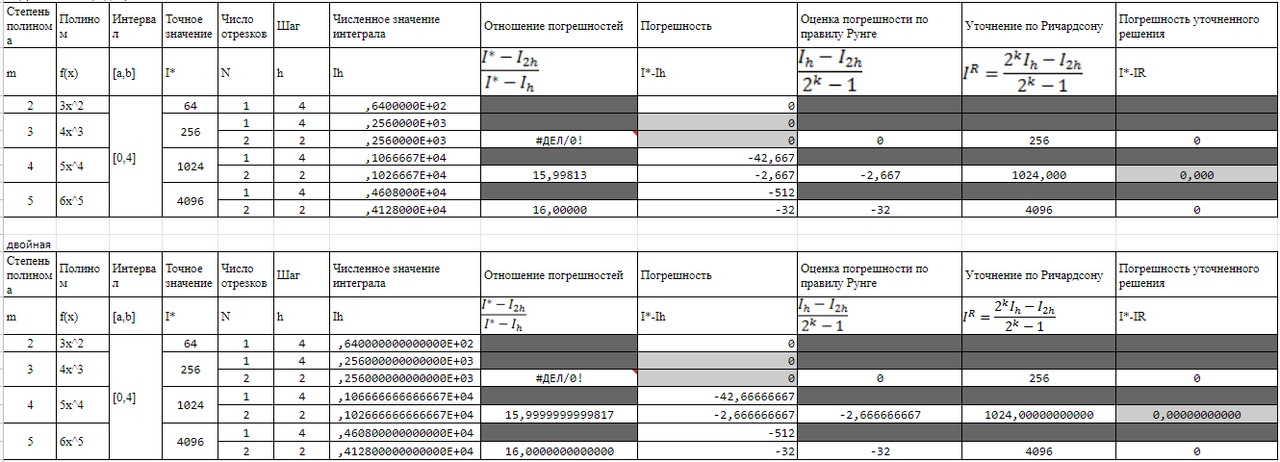
****

1. **Округление по Брадису**

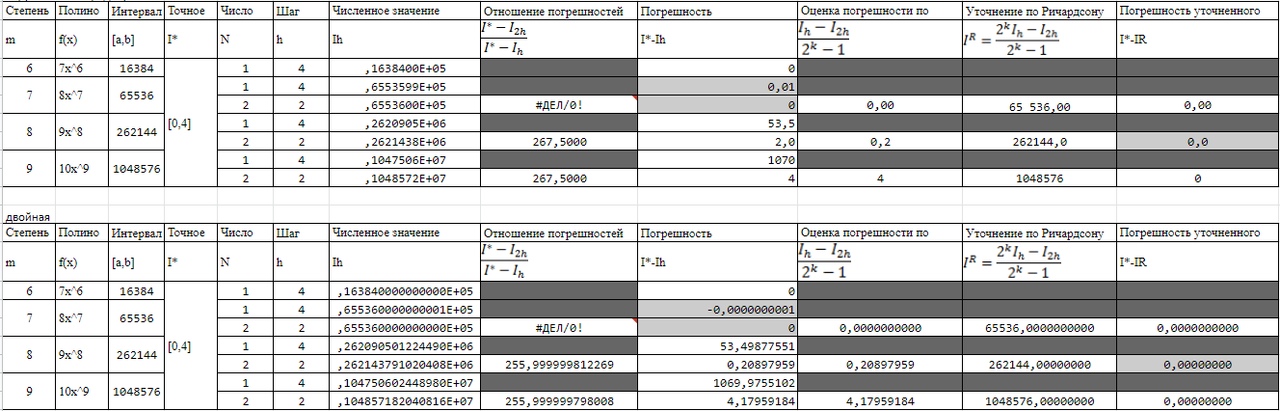
Метод прямоугольников



Метод Симпсона

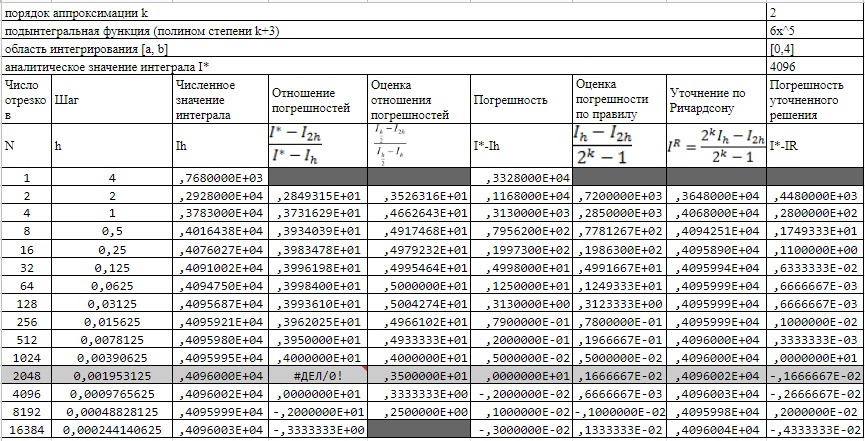


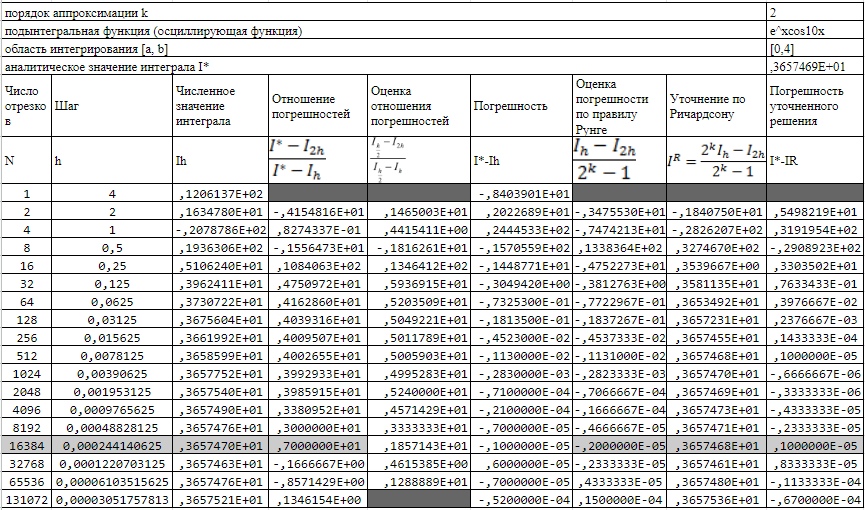
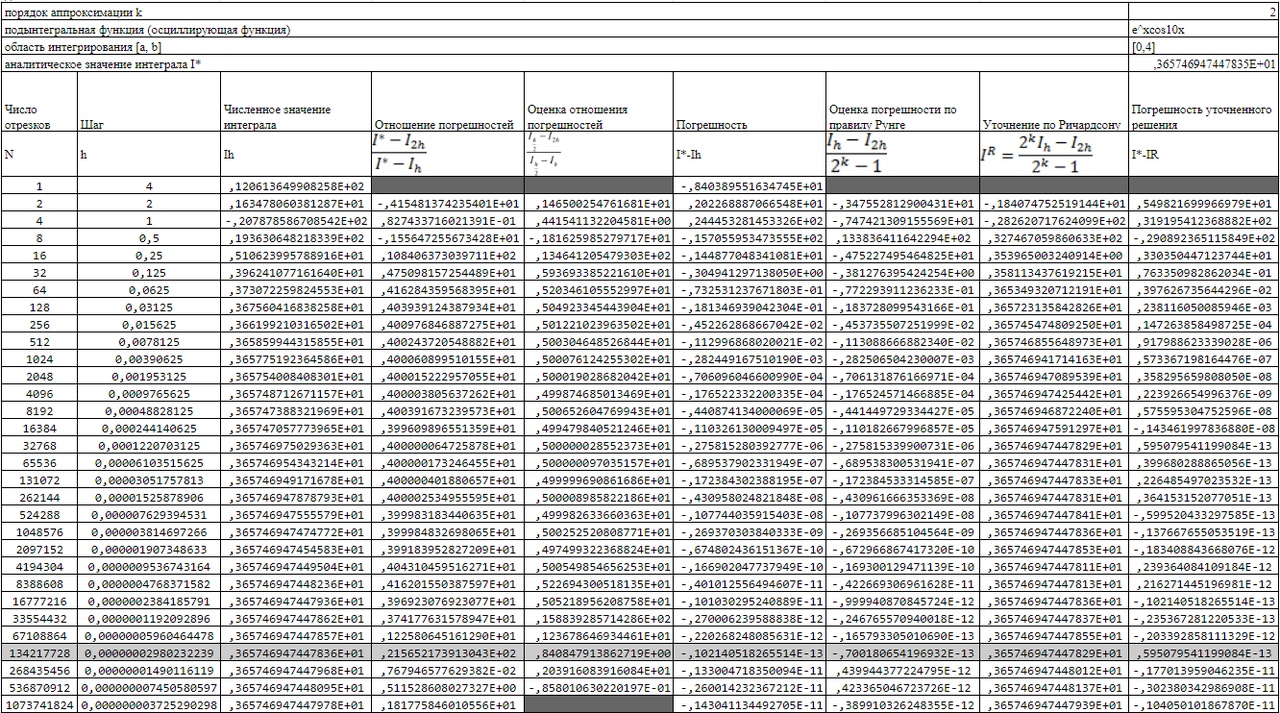
Метод Гаусса-4



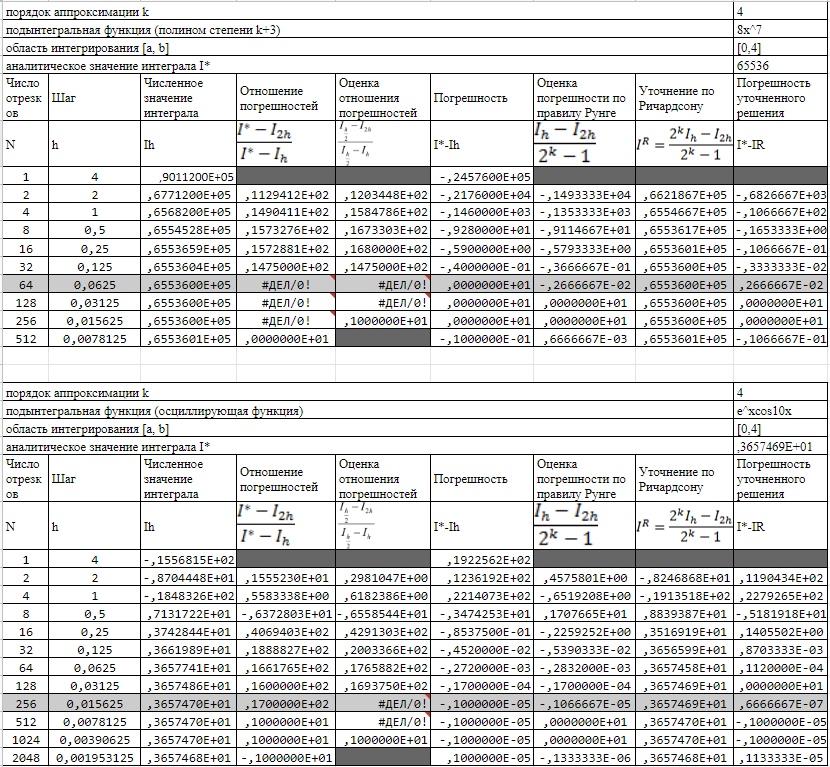
1. **Исследование порядка аппроксимации**

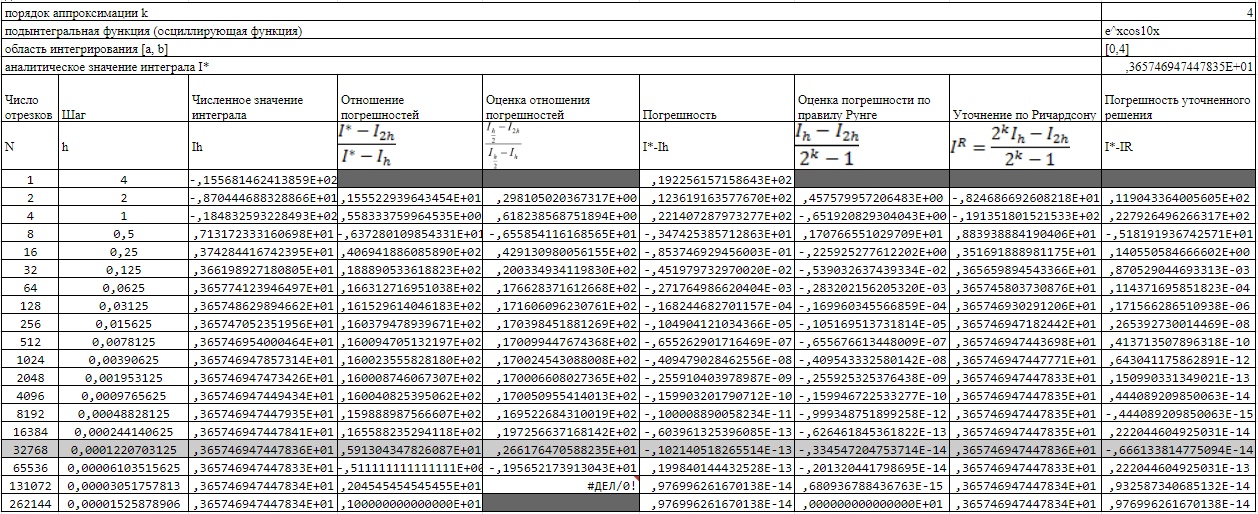
Метод прямоугольника



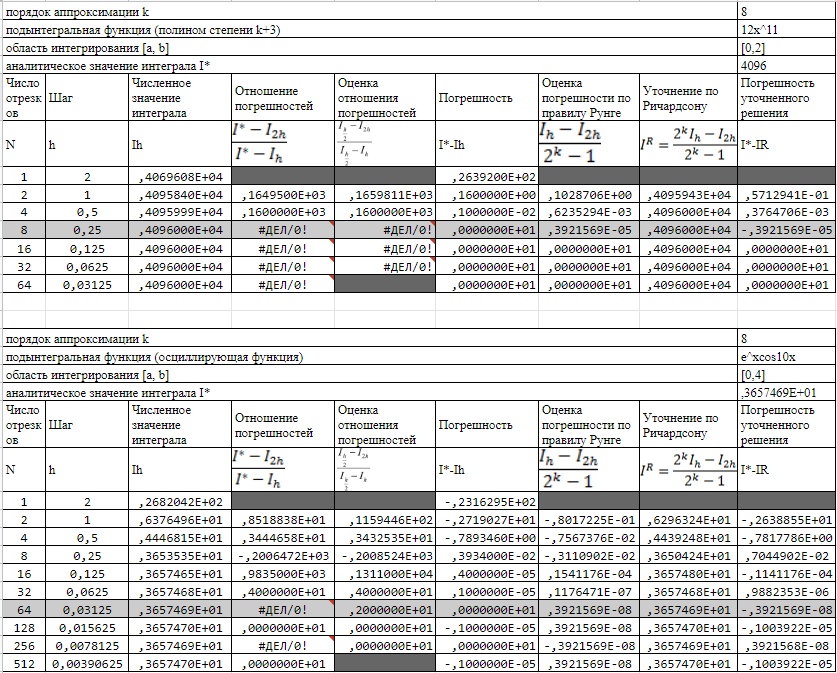
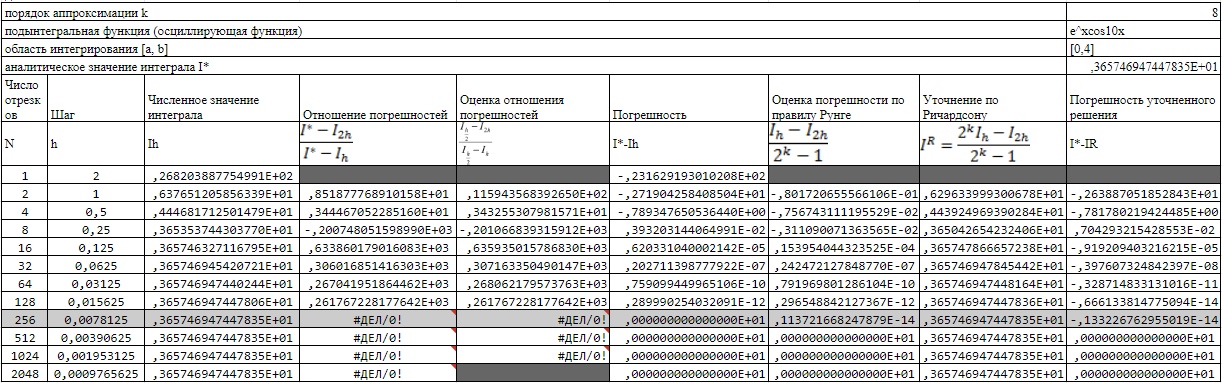
 

Метод Симпсона





Метод Гаусс-4

1. **Текст программы**

**Одинарная точность**

real function f(x)

implicit none

real x

f=exp(x)\*cos(10.\*x)

end

real function g(x,n)

implicit none

real x

integer n

g=(n+1)\*x\*\*n

end

real function rectpoly(n,nn,a,b)

implicit none

real sum1,a,b

real h,g

integer n,nn,i

h=(b-a)/nn

sum1=0.

do i=1,nn

sum1=sum1+h\*g(((2.\*a+((i-1)+i)\*h)/2.),n)

enddo

rectpoly=sum1

end

real function rectoscil(nn,a,b)

implicit none

real sum1,a,b

real h,f

integer nn,i

h=(b-a)/nn

sum1=0.

do i=1,nn

sum1=sum1+h\*f((2.\*a+((i-1)+i)\*h)/2.)

enddo

rectoscil=sum1

return

end

real function simppoly(n,nn,a,b)

implicit none

real sum1,sum2,a,b

real h,g

integer n,nn,i

h=(b-a)/nn

sum2=0.

sum1=0.

do i=1,nn

sum1=sum1+g((a+(i-1)\*h+h/2.),n)

enddo

do i=2,nn

sum2=sum2+g((a+(i-1)\*h),n)

enddo

simppoly=h\*(g(a,n)+g(b,n)+4.\*sum1+2.\*sum2)/6.

end

real function simposcil(nn,a,b)

implicit none

real sum1,sum2,a,b

real h,f

integer nn,i

h=(b-a)/nn

sum1=0.

sum2=0.

do i=1,nn

sum1=sum1+f(a+(i-1)\*h+h/2.)

enddo

do i=2,nn

sum2=sum2+f(a+(i-1)\*h)

enddo

simposcil=h\*(f(a)+f(b)+4.\*sum1+2.\*sum2)/6.

return

end

real function gausspoly(n,nn,a,b)

implicit none

real sum1,sum2,q,x,a,b

real h,g

dimension q(4),x(4)

integer n,nn,j,k

h=(b-a)/nn

sum2=0.

x(1)=-0.8611363

x(2)=-0.3399810

x(3)=0.3399810

x(4)=0.8611363

q(1)=0.3478548

q(2)=0.6521451

q(3)=0.6521451

q(4)=0.3478548

do j=1,4

sum1=0.

do k=0,nn-1

sum1=sum1+h\*g((a+h\*(k)+a+h\*(k+1))/2.+x(j)\*h/2.,n)

enddo

sum2=sum2+q(j)\*sum1

enddo

gausspoly=sum2/2.

return

end

real function gaussoscil(nn,a,b)

implicit none

real sum1,sum2,q,x,a,b

real h,f

dimension q(4),x(4)

integer nn,j,k

h=(b-a)/nn

sum2=0.

x(1)=-0.8611363

x(2)=-0.3399810

x(3)=0.3399810

x(4)=0.8611363

q(1)=0.3478548

q(2)=0.6521451

q(3)=0.6521451

q(4)=0.3478548

do j=1,4

sum1=0.

do k=0,nn-1

sum1=sum1+h\*f((a+h\*k+a+h\*(k+1))/2.+x(j)\*h/2.)

enddo

sum2=sum2+q(j)\*sum1

enddo

gaussoscil=sum2/2.

return

end

program main

implicit none

integer m1,nn,m2,n

real a,b,rectpoly,rectoscil,simppoly,simposcil,gausspoly,

\*gaussoscil

1 print\*, 'Select method:'

print\*, '<1> Rectangle'

print\*, '<2> Simpson'

print\*, '<3> Gaus-4'

print\*, '<4> Exit'

read(\*,\*) m1

if(m1.eq.4) goto 10

if(m1.eq.1)then

print\*, 'Method Rectangle'

print\*, 'Enter quantity of segments:'

read(\*,\*) nn

print\*, 'Enter begin and end:'

read(\*,\*) a,b

2 print\*, 'Select:'

print\*, '1) Polynomial'

print\*, '2) Oscillating'

read(\*,\*) m2

if(m2.lt.1.or.m2.gt.2) goto 2

if(m2.eq.1)then

print\*, 'Enter degree of polynomial:'

read(\*,\*)n

open(10,file='rectpoly.txt',status='unknown',err=5)

write(10,300) nn,a,b,rectpoly(n,nn,a,b)

write(\*,300) nn,a,b,rectpoly(n,nn,a,b)

close(10)

endif

if(m2.eq.2)then

open(10,file='rectoscil.txt',status='unknown',err=5)

write(10,300) nn,a,b,rectoscil(nn,a,b)

write(\*,300) nn,a,b,rectoscil(nn,a,b)

close(10)

endif

endif

if(m1.eq.2)then

print\*, 'Method Simpson'

print\*, 'Enter quantity of segments:'

read(\*,\*) nn

print\*, 'Enter begin and end:'

read(\*,\*) a,b

3 print\*, 'Select:'

print\*, '1) Polynomial'

print\*, '2) Oscillating'

read(\*,\*) m2

if(m2.lt.1.or.m2.gt.2) goto 3

if(m2.eq.1)then

print\*, 'Enter degree of polynomial:'

read(\*,\*)n

open(10,file='simppoly.txt',status='unknown',err=5)

write(10,300) nn,a,b,simppoly(n,nn,a,b)

write(\*,300) nn,a,b,simppoly(n,nn,a,b)

close(10)

endif

if(m2.eq.2)then

open(10,file='simposcil.txt',status='unknown',err=5)

write(10,300) nn,a,b,simposcil(nn,a,b)

write(\*,300) nn,a,b,simposcil(nn,a,b)

close(10)

endif

endif

if(m1.eq.3)then

print\*, 'Method Gaus-4'

print\*, 'Enter quantity of segments:'

read(\*,\*) nn

print\*, 'Enter begin and end:'

read(\*,\*) a,b

4 print\*, 'Select:'

print\*,'1) Polynomial'

print\*, '2) Oscillating'

read(\*,\*) m2

if(m2.lt.1.or.m2.gt.2) goto 4

if(m2.eq.1)then

print\*, 'Enter degree of polynomial:'

read(\*,\*) n

open(10,file='gausspoly.txt',status='unknown',err=5)

write(10,300) nn,a,b,gausspoly(n,nn,a,b)

write(\*,300) nn,a,b,gausspoly(n,nn,a,b)

close(10)

endif

if(m2.eq.2)then

open(10,file='gaussoscil.txt',status='unknown',err=5)

write(10,300) nn,a,b,gaussoscil(nn,a,b)

write(\*,300) nn,a,b,gaussoscil(nn,a,b)

close(10)

endif

endif

print\*, 'Completed.'

pause

goto 1

300 format('Quantity of segments = 'I8/'a = 'E13.7/'b = 'E13.7/

,'Integral = 'E13.7)

301 format('Degree of polynomial = 'I8)

5 print\*, 'Error writing file!'

goto 1

10 end

**Двойная точность**

real\*8 function f(x)

implicit none

real\*8 x

f=dexp(x)\*dcos(10.D00\*x)

end

real\*8 function g(x,n)

implicit none

real\*8 x

integer n

g=(n+1)\*x\*\*n

end

real\*8 function rectpoly(n,nn,a,b)

implicit none

real\*8 sum1,a,b

real\*8 h,g

integer n,nn,i

h=(b-a)/nn

sum1=0.D00

do i=1,nn

sum1=sum1+h\*g(((2.D00\*a+((i-1)+i)\*h)/2.D00),n)

enddo

rectpoly=sum1

end

real\*8 function rectoscil(nn,a,b)

implicit none

real\*8 sum1,a,b

real\*8 h,f

integer nn,i

h=(b-a)/nn

sum1=0.D00

do i=1,nn

sum1=sum1+h\*f((2.D00\*a+((i-1)+i)\*h)/2.D00)

enddo

rectoscil=sum1

return

end

real\*8 function simppoly(n,nn,a,b)

implicit none

real\*8 sum1,sum2,a,b

real\*8 h,g

integer n,nn,i

h=(b-a)/nn

sum2=0.D00

sum1=0.D00

do i=2,nn

sum2=sum2+g((a+(i-1)\*h),n)

enddo

do i=1,nn

sum1=sum1+g((a+(i-1)\*h+h/2.D00),n)

enddo

simppoly=h\*(g(a,n)+g(b,n)+4.D00\*sum1+2.D00\*sum2)/6.D00

end

real\*8 function simposcil(nn,a,b)

implicit none

real\*8 sum1,sum2,a,b

real\*8 h,f

integer nn,i

h=(b-a)/nn

sum1=0.D00

sum2=0.D00

do i=2,nn

sum2=sum2+f(a+(i-1)\*h)

enddo

do i=1,nn

sum1=sum1+f(a+(i-1)\*h+h/2.D00)

enddo

simposcil=h\*(f(a)+f(b)+4.D00\*sum1+2.D00\*sum2)/6.D00

return

end

real\*8 function gausspoly(n,nn,a,b)

implicit none

real\*8 sum1,sum2,q,x,a,b

real\*8 h,g

dimension q(4),x(4)

integer n,nn,j,k

h=(b-a)/nn

sum2=0.

x(1)=-0.861136311594053D00

x(2)=-0.339981043584856D00

x(3)=0.339981043584856D00

x(4)=0.861136311594053D00

q(1)=0.347854845137454D00

q(2)=0.652145154862546D00

q(3)=0.652145154862546D00

q(4)=0.347854845137454D00

do j=1,4

sum1=0.D00

do k=0,nn-1

sum1=sum1+h\*g((a+h\*(k)+a+h\*(k+1))/2.D00+x(j)\*h/2.D00,n)

enddo

sum2=sum2+q(j)\*sum1

enddo

gausspoly=sum2/2.D00

return

end

real\*8 function gaussoscil(nn,a,b)

implicit none

real\*8 sum1,sum2,q,x,a,b

real\*8 h,f !4

dimension q(4),x(4)

integer nn,j,k

h=(b-a)/nn

sum2=0.D00

x(1)=-0.861136311594053D00

x(2)=-0.339981043584856D00

x(3)=0.339981043584856D00

x(4)=0.861136311594053D00

q(1)=0.347854845137454D00

q(2)=0.652145154862546D00

q(3)=0.652145154862546D00

q(4)=0.347854845137454D00

do j=1,4

sum1=0.0D00

do k=0,nn-1

sum1=sum1+h\*f((a+h\*k+a+h\*(k+1))/2.D00+x(j)\*h/2.D00)

enddo

sum2=sum2+q(j)\*sum1

enddo

gaussoscil=sum2/2.D00

return

end

program main

implicit none

integer m1,nn,m2,n

real\*8 a,b,rectpoly,rectoscil,simppoly,simposcil,gausspoly,

\*gaussoscil

1 print\*, 'Select method:'

print\*, '<1> Rectangle'

print\*, '<2> Simpson'

print\*, '<3> Gaus-4'

print\*, '<4> Exit'

read(\*,\*) m1

if(m1.eq.4) goto 10

if(m1.eq.1)then

print\*, 'Method Rectangle'

print\*, 'Enter quantity of segments:'

read(\*,\*) nn

print\*, 'Enter begin and end:'

read(\*,\*) a,b

2 print\*, 'Select:'

print\*, '1) Polynomial'

print\*, '2) Oscillating'

read(\*,\*) m2

if(m2.lt.1.or.m2.gt.2) goto 2

if(m2.eq.1)then

print\*, 'Enter degree of polynomial:'

read(\*,\*)n

open(10,file='rectpoly.txt',status='unknown',err=5)

write(10,300) nn,a,b,rectpoly(n,nn,a,b)

write(\*,300) nn,a,b,rectpoly(n,nn,a,b)

close(10)

endif

if(m2.eq.2)then

open(10,file='rectoscil.txt',status='unknown',err=5)

write(10,300) nn,a,b,rectoscil(nn,a,b)

write(\*,300) nn,a,b,rectoscil(nn,a,b)

close(10)

endif

endif

if(m1.eq.2)then

print\*, 'Method Simpson'

print\*, 'Enter quantity of segments:'

!количество сегментов разбиения отрезка

read(\*,\*) nn

print\*, 'Enter begin and end:'

read(\*,\*) a,b

3 print\*, 'Select:'

print\*, '1) Polynomial'

print\*, '2) Oscillating'

read(\*,\*) m2

if(m2.lt.1.or.m2.gt.2) goto 3

if(m2.eq.1)then

print\*, 'Enter degree of polynomial:'

read(\*,\*)n

open(10,file='simppoly.txt',status='unknown',err=5)

write(10,300) nn,a,b,simppoly(n,nn,a,b)

write(\*,300) nn,a,b,simppoly(n,nn,a,b)

close(10)

endif

if(m2.eq.2)then

open(10,file='simposcil.txt',status='unknown',err=5)

write(10,300) nn,a,b,simposcil(nn,a,b)

write(\*,300) nn,a,b,simposcil(nn,a,b)

close(10)

endif

endif

if(m1.eq.3)then

print\*, 'Method Gaus-4'

print\*, 'Enter quantity of segments:'

read(\*,\*) nn

print\*, 'Enter begin and end:'

read(\*,\*) a,b

4 print\*, 'Select:'

print\*,'1) Polynomial'

print\*, '2) Oscillating'

read(\*,\*) m2

if(m2.lt.1.or.m2.gt.2) goto 4

if(m2.eq.1)then

print\*, 'Enter degree of polynomial:'

read(\*,\*) n

open(10,file='gausspoly.txt',status='unknown',err=5)

write(10,300) nn,a,b,gausspoly(n,nn,a,b)

write(\*,300) nn,a,b,gausspoly(n,nn,a,b)

close(10)

endif

if(m2.eq.2)then

open(10,file='gaussoscil.txt',status='unknown',err=5)

write(10,300) nn,a,b,gaussoscil(nn,a,b)

write(\*,300) nn,a,b,gaussoscil(nn,a,b)

close(10)

endif

endif

print\*, 'Completed.'

pause

goto 1

300 format('Quantity of segments = 'I8/'a = 'E21.15/'b = 'E21.15/

,'Integral = 'E21.15)

301 format('Degree of polynomial = 'I8)

5 print\*, 'Error writing file!'

goto 1

10 end

1. **Вывод**

В ходе выполнения практического задания были изученные методы численного интегрирования, была произведена оценка порядка точности, оценка погрешности по правилу Рунге и уточнение по Ричардсону. По таблицам из задания 3 (верификация программы с помощью степени полинома) видно, что численное значение интеграла является точным при степени полинома m и m-1, где m – теоретическое значение порядка точности, при любом числе отрезков интегрирования. Дальше при увеличении степени полинома и том же числе отрезков погрешность увеличивается. Также хорошо видно, что уточнение по Ричардсону имеет порядок точности на один больше. Однако, в некоторых ячейках погрешности уточненного значения можно увидеть числа малого порядка(из-за представления чисел), эти неточности пропадают после округления по Брадису.

При исследовании полинома степени k+3, где k – порядок аппроксимации, получилось следующее: во всех методах при каждом новом шаге уменьшается порядок погрешностей. По таблицам видно, что первые два метода не всегда могут дать максимально точный ответ, а также после вывода значения с наименьшей погрешностью при увеличении шага погрешность начинает увеличиваться. При методе Гаусса -4 значение интеграла является точным и погрешность минимальна или вовсе отсутствует, вследствие чего оценка погрешности по Рунге минимальна. Погрешность имеет точность на порядок ниже, чем оценка погрешности по Рунге.

Также можно заметить, что почти во всех случаях уточнение по Ричардсону дает точное значение на несколько шагов раньше.