Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет «Московский институт электронной техники»

Лабораторная работа «Интегрирование функций. Формулы трапеций, Симпсона»

Работу выполнил
Учащийся группы ПИН-33
Карпеченков Михаил Владимирович
Под руководством
Васекина Бориса Васильевича

Москва 2023

① Задайте функцию $f(x) = x^3$ на отрезке $[0, 1]$. Очевидно, определённый интеграл от функции $f(x)$ на этом отрезке равен $\frac{1}{4}$. Напишите программу, вычисляющую значение интеграла по формулам трапеций и Симпсона. Какую максимальную теоретическую ошибку мы при этом допускаем? Найдите реальное значение погрешности (абсолютное значение разности между теоретическим и аналитическим решением). Почему при вычислении интеграла по формуле Симпсона от данной функции ошибка равна нулю? Какие бы получились значения погрешностей для квадратичной и линейной функций (предположите и проведите численный эксперимент для $f_2(x) = x^2$, $f_1(x) = x/2$ на отрезке $[0, 1]$).

```
function [res] = integrateByTrapezoid(h,F)
res=0;
steps=0;
for i=2:1:length(F)
    res=res+(F(i)+F(i-1))*h/2;
    steps=steps+1;
end
steps
end


function [res] = integrateBySimpson(h,F)
res=0;
for i=3:2:length(F)
    res=res+h/6*(F(i-2)+F(i)+4*F(i-1));
end
end


clear; clc;
format long; syms x;
f1=matlabFunction(x^3);
f2=matlabFunction(x^2);
f3=matlabFunction(x/2);
f1_2=matlabFunction(diff(x^3,2));
f2_2=matlabFunction(diff(x^2,2));
f3_2=matlabFunction(diff(x/2,2));
f1_4=matlabFunction(diff(x^3,4));
f2_4=matlabFunction(diff(x^2,4));
f3_4=matlabFunction(diff(x/2,4));
step=10^-3; b=1; a=0;
X=a:step:b;
```

```
F1=f1(X);
int1=integrateByTrapezoid(step,F1)
MaxTheoryTrapezoidError=step^2*(b-
a)/12*feval(f1_2,fminbnd(matlabFunction(-diff(x^3,2)),a-1, b+1))
MaxPracticeTrapezoidError=abs(int1-0.25)
X=a:step/2:b;
F1=f1(X);
int2=integrateBySimpson(step,F1)
y=matlabFunction(-diff(x^3,4))
MaxTheorySimpsonError=step^4*(b-a)/2880*0
MaxPracticeSimpsonError=abs(int2-0.25)
X=a:step:b;
F1=f1(X);

F2=f2(X);
int1=integrateByTrapezoid(step,F2)
MaxTheoryTrapezoidError=step^2*(b-a)/12*2
MaxPracticeTrapezoidError=abs(int1-1/3)
X=a:step/2:b;
F2=f2(X);
int2=integrateBySimpson(step,F2)
MaxTheorySimpsonError=step^4*(b-a)/2880*0
MaxPracticeSimpsonError=abs(int2-1/3)
X=a:step:b;
F2=f2(X);

F3=f3(X);
int1=integrateByTrapezoid(step,F3)
MaxTheoryTrapezoidError=step^2*(b-a)/12*0
MaxPracticeTrapezoidError=abs(int1-0.25)
X=a:step/2:b;
F3=f3(X);
int2=integrateBySimpson(step,F3)
MaxTheorySimpsonError=step^4*(b-a)/2880*0
MaxPracticeSimpsonError=abs(int2-0.25)
X=a:step:b;
F3=f3(X);
```

```
steps =

    1000

int1 =

  0.250000250000000

MaxTheoryTrapezoidError =

   9.999788056487483e-07

MaxPracticeTrapezoidError =
```

```
    2.499999998128999e-07

int2 =

  0.250000000000000

y =

 function_handle with value:

   @()0.0

MaxTheorySimpsonError =

    0

MaxPracticeSimpsonError =

    1.110223024625157e-16

steps =

      1000

int1 =

  0.333333500000000

MaxTheoryTrapezoidError =

    1.666666666666667e-07

MaxPracticeTrapezoidError =

    1.666666665234295e-07

int2 =

  0.333333333333333

MaxTheorySimpsonError =

    0

MaxPracticeSimpsonError =

    0

steps =

      1000

int1 =

  0.250000000000000

MaxTheoryTrapezoidError =

    0
```

MaxPracticeTrapezoidError =

5.551115123125783e-17

int2 =

0.250000000000000

MaxTheorySimpsonError =

0

MaxPracticeSimpsonError =

1.110223024625157e-16

② Используя соотношение $\int_0^1 \frac{1}{1+x^2}dx = \mathrm{arctg}(1)$ найдите значение числа π с точностью 10^{-6}. В данном задании в процессе вычислений нельзя использовать встроенную константу `pi` для определения величины шага. Из каких соображений выбирался шаг для получения указанной точности?

```
clear; clc;
format long; syms x;
f=matlabFunction(1/(1+x^2));
step=10^-3; b=1; a=0;
X=a:step:b;
F=f(X);
int1=4*integrateByTrapezoid(step,F)
X=a:step/2:b;
F=f(X);
int2=4*integrateBySimpson(step,F)
pi-int1
pi-int2
```

```
steps =
        1000
int1 =
   3.141592486923128
int2 =
   3.141592653589791
ans =
     1.666666653576954e-07
ans =
     1.776356839400250e-15
```

③ Реализовать предыдущее задание, определяя точность методом Рунге. При численном вычислении интегралов последовательно с шагами h и $h/2$ можно сократить число арифметических операций. Заметим, что приближённое значение интеграла $I_{h/2}$ есть сумма, часть слагаемых которой возможно уже участвовало при вычислении I_h. Поэтому можно получить $I_{h/2}$, используя числовое значение I_h. Это позволяет избежать повторного суммирования части слагаемых[5].

```
function [ShNext] = byMethodRunge(a,b,counter,f,eps)
X=a:(b-a)/(counter-1):b;
h=X(2)-X(1);
F=f(X);
s = sum(F);
Sh=h*(s-1/2*F(1)-1/2*F(length(F)));
counter = counter*2-1;
X=a:(b-a)/(counter-1):b;
h=h/2;
F=f(X);
ShNext=Sh/2+h*(sum(F)-s);
step=0;
while(abs(Sh-ShNext)/3>eps)
    Sh=ShNext;
    s=sum(F);
    counter = counter*2-1;
    X=a:(b-a)/(counter-1):b;
    h=h/2;
    F=f(X);
    ShNext=Sh/2+h*(sum(F)-s);
    step=step+1;
end
step
ShNext;
end
```

```
clear; clc;
format long; syms x;
f=matlabFunction(1/(1+x^2));
h=10^-1; counter=1/h+1; b=1; a=0; eps = 10^-9;
X=a:h:b;
F=f(X);
int=4*byMethodRunge(a,b,counter,f,eps)
pi-int
```

step =

```
    9

int =

  3.141592652000346

ans =

    1.589446796401717e-09
```