Название работы : Численное интегрирование

Оборудование : Ноутбук hp-pavilion ,Python3 , Microsoft office word.

Постановка задачи: Составить программу, которая реализует методы численного интегрирования с постоянным и переменным шагом для интеграла

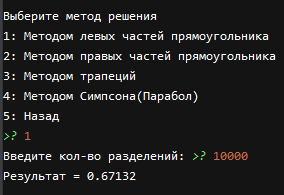
Ссылка: https://github.com/Meresnes/meresnes.github.io/tree/master/Vich\_mat/integrirovanie

Код:

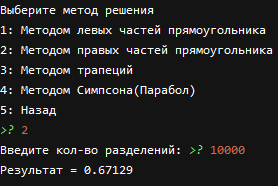
import math  
  
class find\_integral:  
 *#Вычисление интегралов* def \_\_init\_\_(self,a,b,c):  
 self.a = a  
 self.b = b  
 self.c = c  
  
 *#Выбор задания для вычисления* def selected\_task(self,x):  
 if self.c == 1:  
 dx = math.sqrt(x + 2)  
 if self.c == 2:  
 dx = (x - 1)  
 if self.c == 3:  
 dx = (math.e\*\*((-x)\*\*2))  
 if self.c == 4:  
 dx = (math.sin(x)\*\*2)  
 return dx  
 *#Метод прямоугольникво левых частей* def rectangle\_left(self):  
 n = int(input('Введите кол-во разделений: '))  
 s = float(0)  
 h = ((self.b - self.a) / n)  
 x = self.a  
  
  
 while (x <= self.b - h):  
 s += 1 / (self.selected\_task(x))  
 x += h  
  
 result = float(h \* s)  
 print('Результат = {:.5}'.format(result))  
  
 *# Метод прямоугольникво правых частей* def rectangle\_right(self):  
 n = int(input('Введите кол-во разделений: '))  
 s = float(0)  
 h = ((self.b - self.a) / n)  
 x = self.a + h  
  
  
 while (x <= self.b):  
 s += 1 / self.selected\_task(x)  
 x += h  
  
 result = float(h \* s)  
 print('Результат = {:.5}'.format(result ))  
  
 *# Методт трапеции* def trapeze(self):  
 n = int(input('Введите кол-во разделений: '))  
 s = float(0)  
 h = ((self.b - self.a) / n)  
 x = self.a + h  
  
  
 while (x <= self.b - h ):  
 s += 1 / self.selected\_task(x)  
 x += h  
  
 result = h \*((math.cos(self.a) + math.cos(self.b)) / 2 + s)  
 print('Результат = {:.5}'.format(result))  
 return result  
  
 *# Метод Симпсона* def simpson(self):  
 n = int(input('Введите кол-во разделений: '))  
 h = ((self.b - self.a) / n)  
 x = self.a + h  
 s1 = 0  
 s2 = 0  
  
  
 while (x <= self.b - h ):  
 s1 += 1 / self.selected\_task(x)  
 x += 2 \* h  
 x = self.a + 2 \* h  
 while (x <= (self.b - 2 \* h)):  
 s2 += 1 / self.selected\_task(x)  
 x += 2 \* h  
  
 result = h / 3 \* ((1 / self.selected\_task(x) + (1 / self.selected\_task(self.b)) + 4 \* s1 + 2 \* s2))  
 print('Результат = {:.5}'.format(result))  
  
class find\_integral\_2:  
 def \_\_init\_\_(self,a,b,c,e):  
 self.a = a  
 self.b = b  
 self.c = c  
 self.e = e  
  
  
 def selected\_task(self, x):  
 if self.c == 1:  
 dx = math.sqrt(x + 2)  
  
 if self.c == 2:  
 dx = (x - 1)  
  
 if self.c == 3:  
 dx = (math.e \*\* ((-x) \*\* 2))  
  
 if self.c == 4:  
 dx = (math.sin(x) \*\* 2)  
  
 return dx  
  
 def trapeze(self,h):  
  
 s = float(0)  
 x = self.a + h  
  
  
 while (x <= self.b - h ):  
 s += (1 / self.selected\_task(x))  
 x += h  
  
 s += h \*((math.cos(self.a) + math.cos(self.b)) / 2 )  
 return s  
  
 def rectangle\_right(self,d):  
  
 h = (self.b - self.a)/d  
 s = float(0)  
 x = self.a + h  
  
 while (x <= self.b):  
 s += 1 / self.selected\_task(x)  
 x += h  
  
 result = float(h \* s)  
 return result  
  
  
 def double\_recount(self):  
  
 h = self.e  
  
 res1 = self.trapeze((self.b-self.a)/h)  
 res2 = self.trapeze((self.b-self.a)/h/2)  
  
 while abs(res2-res1) > self.e:  
 h /= 2  
 res1 = self.trapeze((self.b-self.a)/h)  
 res2 = self.trapeze((self.b-self.a)/h)  
  
 print('Результат = {:.5} '.format(res2/10))  
 return res2  
  
 def second\_algorithm(self):  
  
 h\_v = self.e  
 h\_s = h\_v / 2  
 res1 = self.rectangle\_right((self.b - self.a)/h\_v)  
 res2 = 0  
  
 while abs(res2 - res1) > self.e:  
  
 res1 = self.rectangle\_right((self.b - self.a)/h\_v)  
 h\_v += h\_s  
 h\_d = h\_v / 2  
  
 res2 = self.rectangle\_right((self.b - self.a)/h\_d)  
 h\_v /= 2  
 h\_s /= 2  
  
 print('Результат = {:.5}'.format(res2))  
 return res2  
  
  
def method():  
  
 while (True):  
 print("\n\nВыберите метод решения")  
 print("1: Метод с постоянным шагом")  
 print("2: Метод с переменным шагом")  
 print("3: Выход")  
   
 z = int(input(">>>"))  
 if (z == 1) | (z == 2)| (z == 3):  
 break  
 else:  
 print("\nВы ввели не правильные данные!!!")  
 return z  
def choose(a,b,c):  
 *#Меню выбора метода решения* while True:  
  
 print("\n\nВыберите метод решения")  
 print("1: Методом левых частей прямоугольника")  
 print("2: Методом правых частей прямоугольника")  
 print("3: Методом трапеций")  
 print("4: Методом Симпсона(Парабол)")  
 print("5: Назад")  
  
 abs = find\_integral(a, b, c)  
 z = int(input(">>>"))  
  
 if z == 1:  
 abs.rectangle\_left()  
 break  
 if z == 2:  
 abs.rectangle\_right()  
 break  
 if z == 3:  
 abs.trapeze()  
 break  
 if z == 4:  
 abs.simpson()  
 break  
 if z == 5:  
 return False  
 break  
 else:  
 print("\nВы ввели не правильные данные!!!")  
  
def choose\_2(a,b,c,e):  
 while True:  
  
 print("\n\nВыберите метод решения")  
 print("1: Первый алгоритм")  
 print("2: Второй алгоритм")  
 print("3: Назад")  
  
 abs = find\_integral\_2(a, b, c, e)  
 z = int(input(">>>"))  
  
 if z == 1:  
 abs.double\_recount()  
 break  
 if z == 2:  
 abs.second\_algorithm()  
 break  
 if z == 3:  
 return False  
 break  
 else:  
 print("\nВы ввели не правильные данные!!!")  
 return z  
def menu():  
 *#Меню выбора задачи* while True:  
  
 print("\n1: dx/√(x+2)")  
 print("2: dx/x-1")  
 print("3:(e^(-x^2))\*dx")  
 print("4:(dx/(sin^2 x))")  
 print("5: Назад")  
  
 c = int(input("\nВыберите задачу:"))  
  
 if c == 5:  
 return False  
 break  
 if ((c == 1)|(c == 2)|(c == 3)|(c == 4)):  
 return c  
 break  
 if ((c != 1) & (c != 2) & (c != 3) & (c != 4)& (c != 5)):  
 print("\nВы ввели не правильные данные!!!")  
  
 return c  
  
  
def main():  
 *#Основная функция* while True:  
 z = method()  
  
 if z == 3:  
 break  
  
 if z == 1:  
 c = menu()  
 if (c != False):  
 a = float(input("Введите нижний предел интегрирования:"))  
 b = float(input("Введите верхний предел интегрирования:"))  
 q = choose(a, b, c)  
   
  
  
 if z == 2:  
 c = menu()  
 if (c != False):  
 a = float(input("Введите нижний предел интегрирования:"))  
 b = float(input("Введите верхний предел интегрирования:"))  
 e = float(input("Введите точность:"))  
  
 q = choose\_2(a, b, c, e)  
   
 if (c == False):  
 pass  
  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()

Результаты работы при a = 1, b = 1.6 , dx/(sin^2 x):

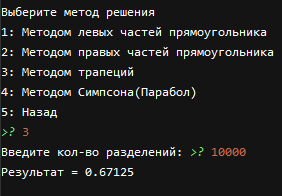
1)



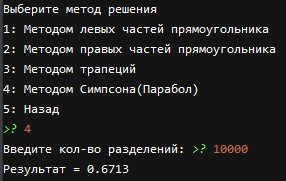
2)



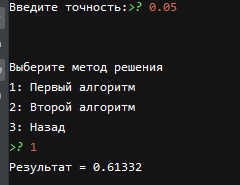
3)



4)



5)



6)

