Задача 7. Требуется найти такие константы a, b, c и d, что квадратура

$$\int_{-1}^{1} f(x)dx = af(-1) + bf(1) + cf'(-1) + df'(1)$$

имеет третью степень точности.

Требуется, чтобы квадратура была точной для $f(x) = P_3(x)$:

$$f(x) = a_o + a_1 x + a_2 x^2 + a_3 x^3$$

$$\Rightarrow \int_{-1}^{1} f(x)dx = a_o \int_{-1}^{1} dx + a_1 \int_{-1}^{1} xdx + a_2 \int_{-1}^{1} x^2 dx + a_3 \int_{-1}^{1} x^3 dx$$

Квадратура точно вычисляет интеграл от $P_3(x)$, когда точно вычисляются интегралы от функций:

$$f(x) = 1, \qquad f(x) = x, \qquad f(x) = x^2, \qquad f(x) = x^3.$$

Производные данных функций соответственно:

$$f'(x) = 0, \qquad f'(x) = 1, \qquad f'(x) = 2x, \qquad f'(x) = 3x^2$$

Полученная система уравнений:

$$\begin{cases} a + b = \int_{-1}^{1} dx = 2 \\ -a + b + c + d = \int_{-1}^{1} xdx = 0 \\ a + b - 2c + 2d = \int_{-1}^{1} x^2 dx = \dfrac{2}{3} \\ -a + b + 3c + 3d = \int_{-1}^{1} x^3 dx = 0 \end{cases} \Leftrightarrow \begin{cases} b = 2 - a \\ c = -d \\ 2b = 2 - c - d \\ 2c - 2d = a + b - \dfrac{2}{3} \end{cases} \Leftrightarrow$$

$$\Leftrightarrow 4 - 2a = 2 - c + c = 2 \Leftrightarrow \begin{cases} a = 1 \\ b = 1 \\ 4c = 2 - \dfrac{2}{3} \\ 4d = -2 + \dfrac{2}{3} \end{cases} \Leftrightarrow \begin{cases} a = 1 \\ b = 1 \\ c = \dfrac{1}{3} \\ d = -\dfrac{1}{3} \end{cases}$$

Итоговая формула:

$$\int_{-1}^{1} f(x)dx = f(-1) + f(1) + \frac{1}{3} f'(-1) - \frac{1}{3} f'(1)$$

Для проверки правильности была написана программа на Python, которая аналитически вычисляет значения интегралов для случайного набора полиномов от 0 до 4 степени и соответствующие значения квадратуры. Вывод программы:

$$\int_{-1}^{1}\left(-\frac{7}{15}\right)dx=-\frac{14}{15}$$

Квадратура: $-\frac{7}{15}-\frac{7}{15}+\frac{1}{3}\cdot 0-\frac{1}{3}\cdot 0=-\frac{14}{15}$

$$\int_{-1}^{1}\left(-\frac{8}{7}+\frac{1}{5}\cdot x\right)dx=-\frac{16}{7}$$

Квадратура: $-\frac{47}{35}-\frac{33}{35}+\frac{1}{3}\cdot\frac{1}{5}-\frac{1}{3}\cdot\frac{1}{5}=-\frac{16}{7}$

$$\int_{-1}^{1}\left(\frac{5}{2}+\frac{19}{17}\cdot x+\frac{16}{7}\cdot x^2\right)dx=\frac{137}{21}$$

Квадратура: $\frac{873}{238}+\frac{1405}{238}+\frac{1}{3}\cdot\left(-\frac{411}{119}\right)-\frac{1}{3}\cdot\frac{677}{119}=\frac{137}{21}$

$$\int_{-1}^{1}\left(\frac{13}{8}+\frac{15}{8}\cdot x+\frac{2}{7}\cdot x^2-\frac{4}{5}\cdot x^3\right)dx=\frac{289}{84}$$

Квадратура: $\frac{117}{140}+\frac{209}{70}+\frac{1}{3}\cdot\left(-\frac{307}{280}\right)-\frac{1}{3}\cdot\frac{13}{280}=\frac{289}{84}$

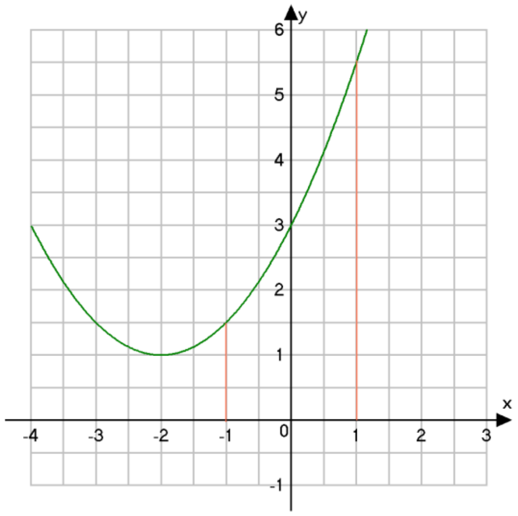
$$\int_{-1}^{1}\left(-\frac{1}{3}+\frac{7}{8}\cdot x-\frac{5}{2}\cdot x^2+20\cdot x^3-11\cdot x^4\right)dx=-\frac{101}{15}$$

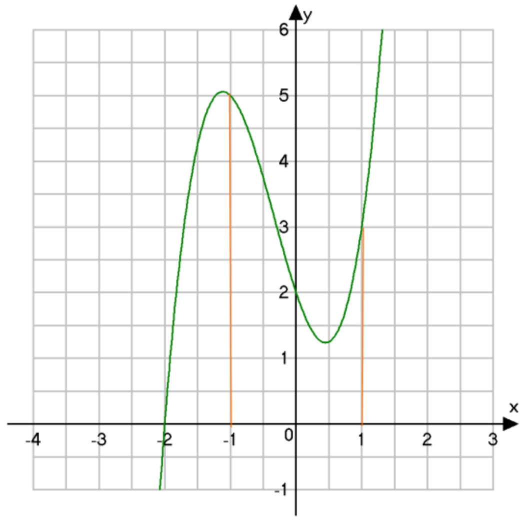
Квадратура: $-\frac{833}{24}+\frac{169}{24}+\frac{1}{3}\cdot\frac{879}{8}-\frac{1}{3}\cdot\frac{95}{8}=5$

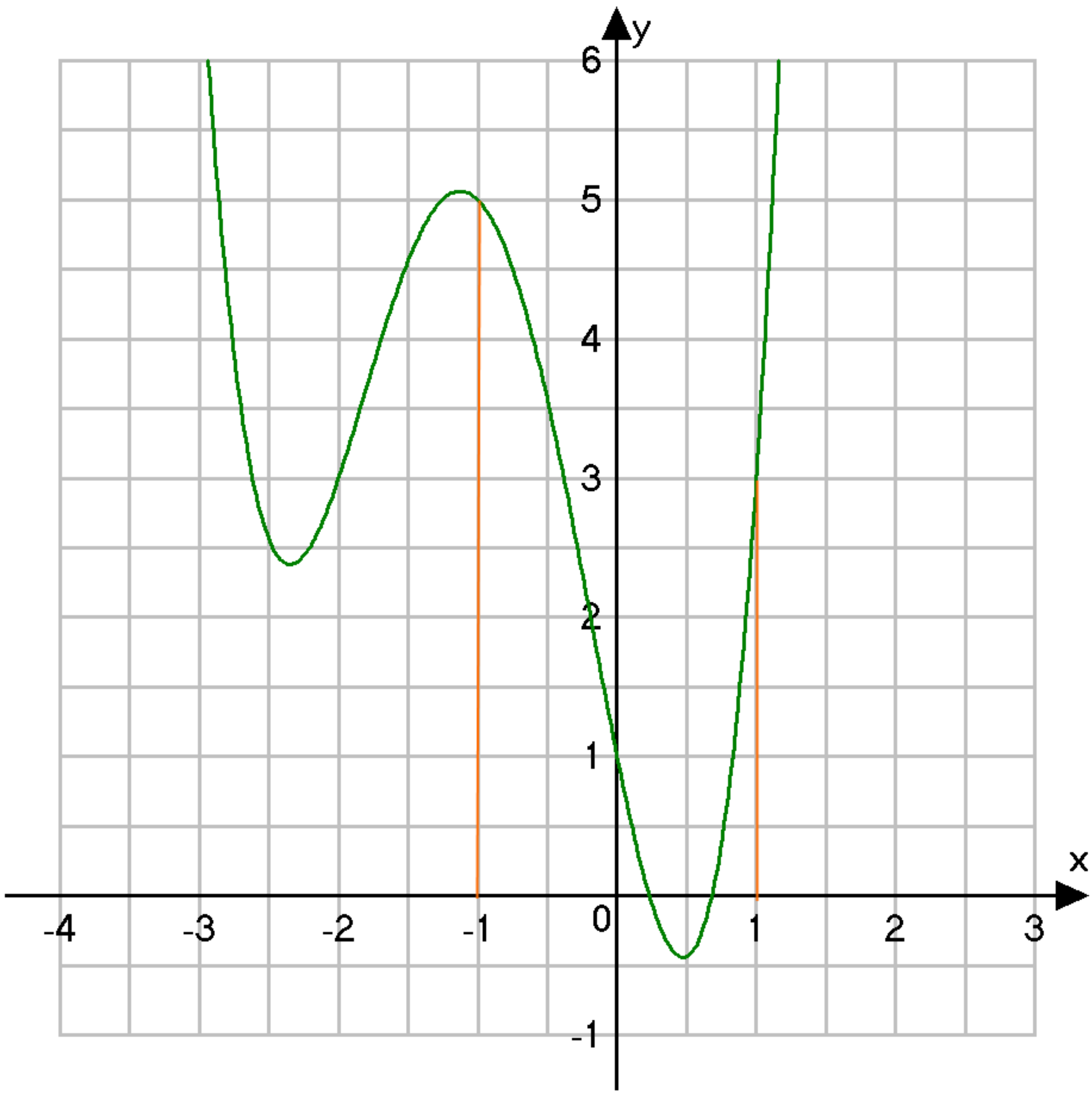
Additional tests were made 49 times:
Polynominals of degree 0: 50/50 match
Polynominals of degree 1: 50/50 match
Polynominals of degree 2: 50/50 match
Polynominals of degree 3: 50/50 match
Polynominals of degree 4: 0/50 match, average relative error: 320.72%

Вывод. Геометрически формулу можно интерпретировать, как аппроксимацию участка двумя константами, равными значениям функции на концах отрезка. Значения производных на концах отрезка говорят о том насколько изменить требуется получившийся результат. Поэтому и коэффициенты на входе и на выходе имеют противоположный знак. В них и заключено соотношение между значениями производных и разницей между площадью фактической площадью и площадью прямоугольника. Для полиномов 4 и выше порядка коэффициенты при производных перестают нести смысловую нагрузку, так как соотношение, описанное выше, становится неверно. И для таких полиномов фактически происходит аппроксимация двумя прямоугольниками и двумя уже мало обоснованными слагаемыми.

Иллюстрации для объяснения.







Листинг.

```
import wordcore as core  #авторская библиотека для работы  с word
import sympy as sp
import random

def rnd(max):
   digits=[]
   for i in range(1,max+1):
      digits.append(i)
      digits.append(-i)
   return sp.Rational(random.choice(digits),random.choice(digits))

def test(power, times=50, name="vychmat"):
   x = sp.symbols("x")
   core.init(debug=True)
   matches=[]
   difference=[]
   for k in range(times): #количество тестов
      for i in range(power+1): #полином степени от нуля до power
         if not k:
            matches.append(0)
            difference.append(0)
         ex=""
         for j in range(i+1): #Генерация полинома
            r=rnd(20)
            ex+=f"+{r}*x^{j}"
         ex=ex.replace("x^0","1")
         ex=ex.replace("x^1", "x")
         ex=core.m_clear(ex)
         original=sp.integrate(ex,(x,-1,1))
         if not k:
            core.add_math(f"int_-1^1▒({ex})dx={original}", mclear=False)
         ex=ex.replace("^","**")
         ex=sp.parse_expr(ex)
         diff=sp.diff(ex,x)
         sumquad=core.m_clear(f"{ex.subs(x,-1)}+{ex.subs(x,1)}+(1/3)*" \
               f"({diff.subs(x,-1)})-(1/3)*({diff.subs(x,1)})")
         quad=sp.simplify(sumquad)
         if not k:
            core.add_math(f"Квадратура: {sumquad}={quad}", mclear=False)
         if quad==original:
            matches[i]+=1
         else:
            difference[i]+=abs((quad-original)/original)
   if times>1:
      if times!=2:
         core.add_text(f"Additional tests were made {times - 1} times:")
      else:
         core.add_text(f"An additional test was made:")
      for i in range(power + 1):
         core.add_text(end=f"Polynominals of degree {i}: {matches[i]}/{times} match")
         if difference[i]:
            core.add_text(f", average relative error: {round(difference[i]/times*100, 2)}%")
         else:
            core.add_text("")
   core.doc_save(name)
test(4, 50)
```