МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

**«Московский технический университет связи и информатики»**

Кафедра «Информатика»

**Отчет по ЛР №4.11**

**по дисциплине**

**«Базовые средства мат. пакетов»**

Выполнил: студент гр. БЭИ2202

Кулешов А. С.

Вариант 14.

Проверил: доц. каф. «Информатика»

Шакин В.Н.

Москва, 2024 г.

1. **Индивидуальное задание**
2. Выбрать самостоятельно две символьные функции (разного типа).
3. Провести исследование математических функций методами дифференциального исчисления в соответствии с примерами п.3.10.
4. Предоставить результаты работы преподавателю и ответить на поставленные вопросы.
5. Оформить отчет по выполненной работе.
6. **Программный код**

from sympy import \*  
  
x = symbols('x')  
  
f = 1.5 \* sin(x) + 0.4 \* cos(2 \* x)  
display(f)  
ff = lambda x: 1.5 \* sin(x) + 0.4 \* cos(2 \* x); ff(x)  
plot((ff(x), (x, -8, 8)), ylim=(-3.2, 2.5), legend=True, title="Предварительный график функции")  
  
z = solve(ff(x), x, dict=true); z  
zr = solveset(ff(x), x); zr  
zz = solveset(ff(x), x, Interval(-8, 8)); zz  
zz = zz.evalf(10); zz  
zz = list(zz); zz  
  
k00 = zz[0]; k01 = zz[1]; k02 = zz[2]; k03 = zz[3]; k04 = zz[4];  
k00.evalf(), k01.evalf(), k02.evalf(), k03.evalf(), k04.evalf()  
ff1 = diff(ff(x), x)  
ff1 = ff1.simplify(); ff1  
print("Первая производная")  
display(ff1)  
  
ans = solveset(ff1, x, Interval(-8, 8)); ans  
ans = ans.evalf(10); ans  
ans = list(ans); ans  
h00 = ans[0]; h01 = ans[1]; h02 = ans[2]; h03 = ans[3]; h04 = ans[4];  
h00, h01, h02, h03, h04  
print("Вторая производная")  
ff2 = diff(ff(x), x, 2); ff2  
display(ff2)  
  
r0 = ff(h00); r1 = ff(h01); r2 = ff(h02); r3 = ff(h03); r4 = ff(h04)  
r0, r1, r2, r3, r4  
plot((ff(x), (x, -8, 8)), ylim=(-3.2, 2.5), legend=True,  
 markers=[{'args': [[h00, h01, h02, h03, h04, k00, k01, k02, k03, k04], [r0, r1, r2, r3, r4, 0, 0, 0, 0, 0], " o"]}])  
x, y, x0 = symbols('x y x0')  
  
y = (x\*\*2 - 16)/(5\*(x + 5)); y  
f = lambda x: (x\*\*2 - 9)/(4 \* (x + 4)); f(x)  
plot((f(x), (x, -15, 15)), ylim=(-15, 5), legend=True, title="Предварительный график функции")  
f(0) # Пересечение с осью ву:  
xx = solve(y, x, dict=True); xx # Пересечение с осью Ох:  
k01 = xx[0][x]; k02 =xx[1][x]  
k01, k02  
  
xxx = solveset(f(x), x); xx # Ось Оу:  
limit(y, x, oo) # Вертикальная асимптома  
k = limit(f(x)/x, x, oo); k # Наклонная асимптото  
b = limit(y-k\*x, x, oo); b  
ya = x/5 -1; ya  
yp1= diff(f(x), x); yp1 #1-я производная от функции f(x)  
yp1 = yp1.simplify(); yp1  
x0 = solve (yp1, dict=True); x0 # Корни ур1 1-ой производна  
x01 = x0[0][x]; x02 = x0[1][x]  
x01, x02 # две критические точки  
  
yp2 = diff(f(x), x, 2); yp2 # 2-я производная от функции f(x)  
yp2 = yp2.simplify(); yp2 # Преобразуем выражение 2-й производной  
exprz= x\*\*3 + 15\*x\*\*2 + 75\*x + 125  
exprz = factor(exprz); exprz  
yp2 = 18/(5\*exprz); yp2  
yp2 = yp2.simplify(); yp2  
yp2.subs(x, 8), yp2.subs(x,-2)  
t1 = f(x01); t2 = f(x02);  
t1, t2  
p1 = plot((f(x), (x, -4.8, 6)), (f(x), (x, -15, 5.2)), (ya, (x, -15, 6)),  
 ylim=(-8, 6), legend=True,  
 markers=[{'args': [[x01, x02, k01, k02], [t1, t2, 0, 0], " o"]}])

1. **Полученный результат**

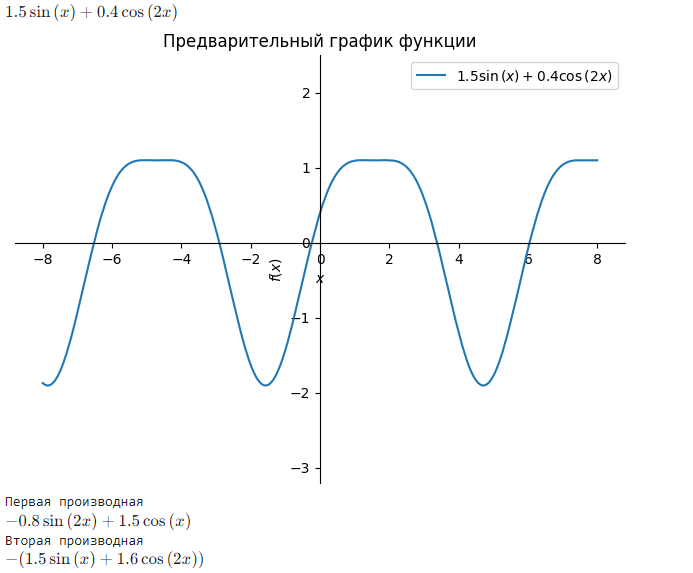
****

Рисунок 1 – Предварительный график первой функции

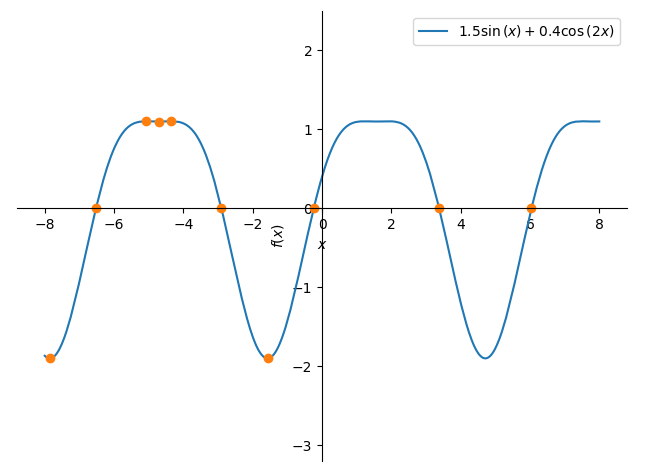


Рисунок 2 – Критические точки и пересечение оси Ox первой функции

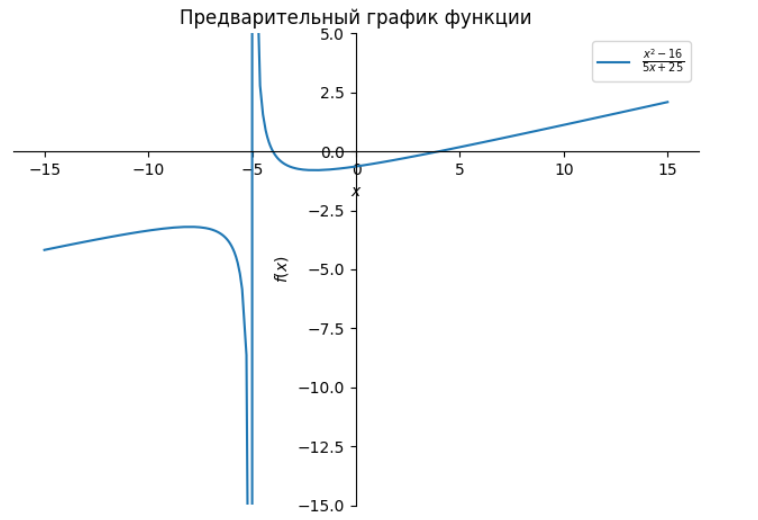


Рисунок 3 – Предварительный график второй функции

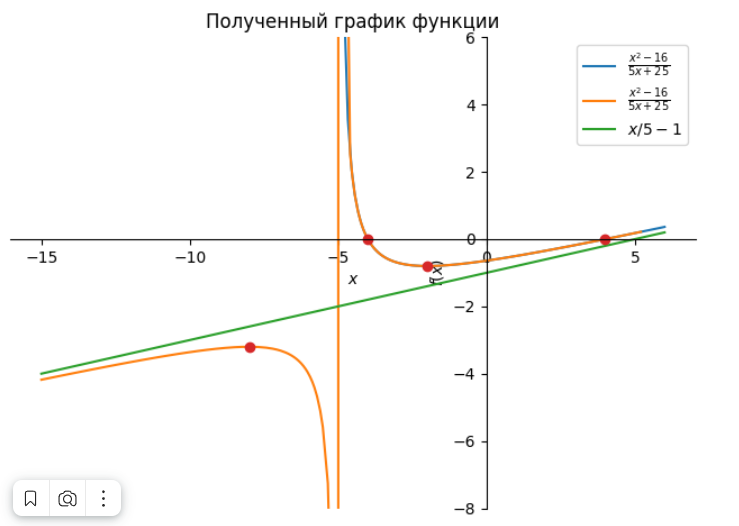


Рисунок 4 – Критические точки и пересечение оси Ox второй функции