

# Бюджетное учреждение высшего образования Ханты-Мансийского автономного округа – Югры

## «СУРГУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

### Политехнический институт

Кафедра прикладной математики

Бондаренко Анна Андреевна

Дисциплина «Алгебра и геометрия»

направление 01.03.02 «Прикладная математика и информатика»

направленность (профиль): «Технологии программирования и анализ данных»

Преподаватель:

Шапошникова Ирина Вадимовна

Студент гр. № 601-31

Бондаренко Анна Андреевна

# Аналитическая геометрия.

## Индивидуальное задание №2.

### Вариант №2.

#### Задание 1

Даны две точки  $A = (1, 2)$  и  $B = (2, 5)$ . Написать программу на языке Python с использованием библиотек `matplotlib` и `numpy`, которая строит рисунок, изображающий прямую, проходящую через эти точки, а также единичный вектор  $n$  нормали этой прямой. На рисунке должны быть подписаны оси  $Ox$  и  $Oy$ , точки  $A$  и  $B$  и вектор  $n$ . Построенный рисунок должен приблизительно иметь вид, показанный на рис. 1. Также программа должна выводить уравнение построенной прямой в виде  $y = kx + b$ .

#### Программное решение

```
import matplotlib.pyplot as plt
import numpy as np
if __name__ == "__main__":
    A = np.array([1, 2])
    B = np.array([2, 5])
    def plot_pic():
        plt.plot([A[0], B[0]], [A[1], B[1]], color='blue')
        plt.scatter(A[0], A[1], color='red', alpha=1)
        plt.scatter(B[0], B[1], color='red', alpha=1)
        normal = np.array([B[1] - A[1], A[0] - B[0]])
        mid_point = (A + B) / 2
        plt.arrow(mid_point[0], mid_point[1], normal[0], normal[1], color='red',
head_width=0.15, head_length=0.3, fc='red', ec='red')
        plt.text(A[0] + 0.4, A[1], 'A', fontsize=12, ha='right')
        plt.text(B[0] - 0.2, B[1], 'B', fontsize=12, ha='right')
        plt.text(mid_point[0] + normal[0]/2, mid_point[1] + normal[1]/2 + 0.2, 'n',
fontsize=12, ha='right')
        plt.xlabel("x", fontsize="xx-large",
fontstyle="italic", family="serif")
        plt.ylabel("y", fontsize="xx-large",
fontstyle="italic", family="serif")
        plt.show()
    plot_pic()
# получение уравнения прямой
```

```

k = (B[1] - A[1]) / (B[0] - A[0])
b = A[1] - k * A[0]
print(f"Уравнение прямой: y = {k}x + ({b})")

```

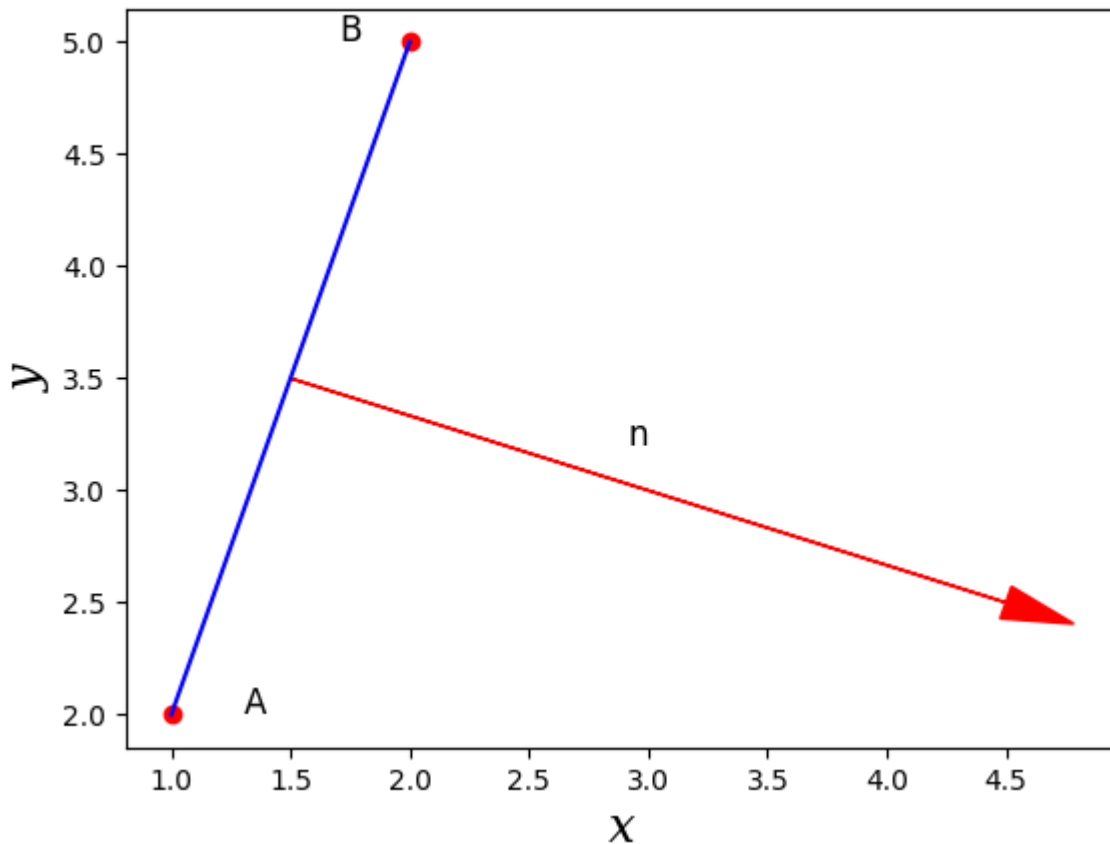


Рис. 1. Иллюстрация решения задачи №1.

```

PS C:\Users\Анна\Documents\GitHub\programming> & C:/Users/Анна/AppData/Local/Microsoft/WindowsApps/python3.11.exe c:/Users/Анна/Documents/GitHub/programming/algeom1/algeom1.py
уравнение прямой: y = 3.0x + (-1.0)
PS C:\Users\Анна\Documents\GitHub\programming>

```

Рис. 2. Вывод в терминале.

## Задание 2

Дана точка  $A = (-4, -3)$  и вектор  $u = (7, 1)$ . Написать программу на языке Python с использованием библиотек `matplotlib` и `numpy`, которая строит рисунок, изображающий прямую, проходящую через точку  $A$  в направлении вектора  $u$ , а также единичный вектор  $n$  нормали этой прямой. На рисунке должны быть подписаны оси  $Ox$  и  $Oy$ , точка  $A$  и векторы  $u$  и  $n$ . Также программа должна выводить уравнение построенной прямой в виде  $y = kx + b$ .

### Программное решение

```

import numpy as np
import matplotlib.pyplot as plt
# Задаем точку A и вектор u
A = np.array([-4, -3])
u = np.array([7, 1])
# Вычисляем координаты конечной точки прямой
B = A + u
# Вычисляем нормальный вектор к прямой
n = np.array([-u[1], u[0]])
# Вычисляем коэффициенты уравнения прямой  $y = kx + b$ 
k = u[1] / u[0]
b = A[1] - k * A[0]
# Создаем фигуру и оси
fig, ax = plt.subplots()
# Рисуем прямую, проходящую через точку A в направлении вектора u
ax.plot([A[0], B[0]], [A[1], B[1]], 'r', label='Прямая')
# Рисуем нормальный вектор
ax.arrow(A[0], A[1], n[0], n[1], head_width=0.2, head_length=0.2, fc='b', ec='b',
label='Нормаль')
# Подписываем оси
ax.set_xlabel('x')
ax.set_ylabel('y')
# Подписываем точку A и векторы u и n
ax.annotate('A', A, textcoords="offset points", xytext=(-6, -14), ha='center')
ax.annotate('u', B, textcoords="offset points", xytext=(8, 4), ha='center')
ax.annotate('n', A + n, textcoords="offset points", xytext=(-10, 10), ha='center')
# Выводим уравнение прямой в терминале
print(f'Уравнение прямой:  $y = \{round(k, 2)\}x + \{round(b, 2)\}$ ')
# Выводим график
plt.legend()
plt.grid(True)
plt.show()

```

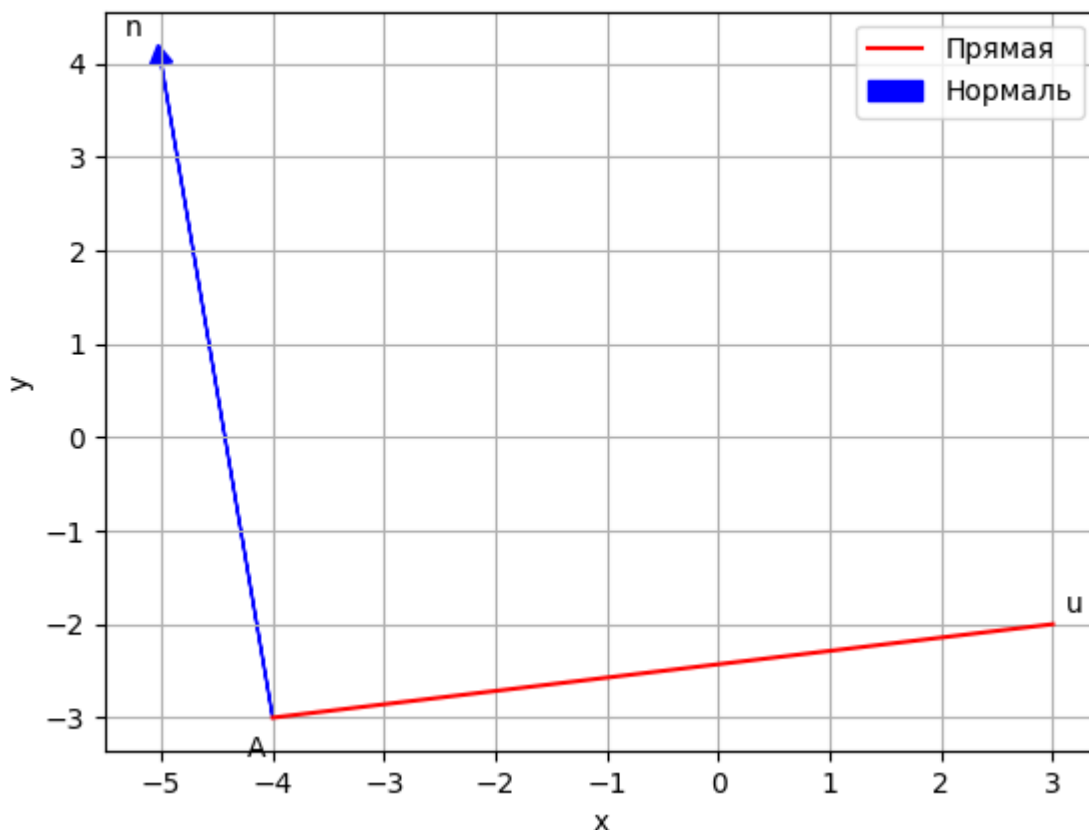


Рис. 3. Иллюстрация решения задачи №2.

```
PS C:\Users\Анна\Documents\GitHub\programming> & C:/Users/Анна/AppData/Local/Microsoft/WindowsApps/python3.11.exe c:/Users/Анна/Documents/GitHub/programming/algeom1/algeom1.py
уравнение прямой: y = 0.14x + -2.43
PS C:\Users\Анна\Documents\GitHub\programming>
```

Рис. 4. Вывод в терминале.

## Задание 3

Дана точка  $A = (-2, 7)$  и вектор  $u = (-1, -1)$ . Написать программу на языке Python с использованием библиотек `matplotlib` и `numpy`, которая строит рисунок, изображающий прямую, проходящую через точку  $A$  перпендикулярно вектору  $u$ . На рисунке должны быть подписаны оси  $Ox$  и  $Oy$ , точка  $A$  и вектор  $u$ . Также программа должна выводить уравнение построенной прямой в виде  $y = kx + b$ .

### Программное решение

```
import numpy as np
import matplotlib.pyplot as plt
# Задаем точку A и вектор u
A = np.array([-2, 7])
u = np.array([-1, -1])
```

```

# Находим коэффициенты прямой
k = -u[0] / u[1]
b = A[1] - k * A[0]
# Создаем массив значений x
x = np.linspace(-10, 10, 100)
# Вычисляем соответствующие значения y
y = k * x + b
# Строим график
plt.plot(x, y, label='y = {}x + {}'.format(k, b))
plt.scatter(A[0], A[1], color='red', label='Точка A')
plt.arrow(A[0], A[1], u[0], u[1], color='blue', width=0.1,
head_width=0.5, length_includes_head=True, label='Вектор u')
plt.xlabel('x')
plt.ylabel('y')
plt.title('Прямая, проходящая через точку A перпендикулярно вектору u')
plt.legend()
plt.grid(True)
plt.show()
# Выводим уравнение прямой
print('Уравнение прямой: y = {}x + {}'.format(k, b))

```

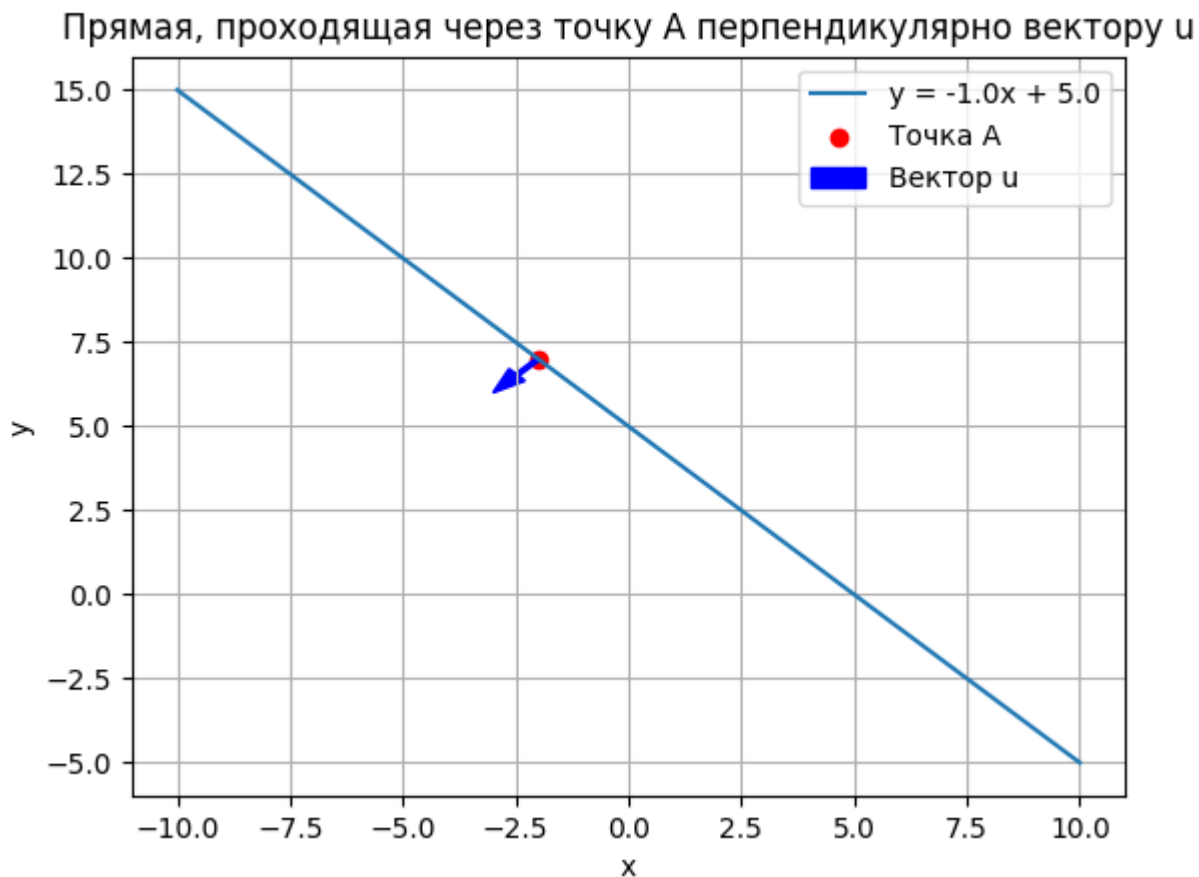


Рис. 5. Иллюстрация решения задачи №3.

```

PS C:\Users\Анна\Documents\GitHub\programming> & C:/Users/Анна/AppData/Local/Microsoft/windowsApps/python3.11.exe c:/Users/Анна/Documents/GitHub/programming/algeom1/algeom1.py
Уравнение прямой: y = -1.0x + 5.0
PS C:\Users\Анна\Documents\GitHub\programming> 

```

Рис. 6. Вывод программы в терминале.