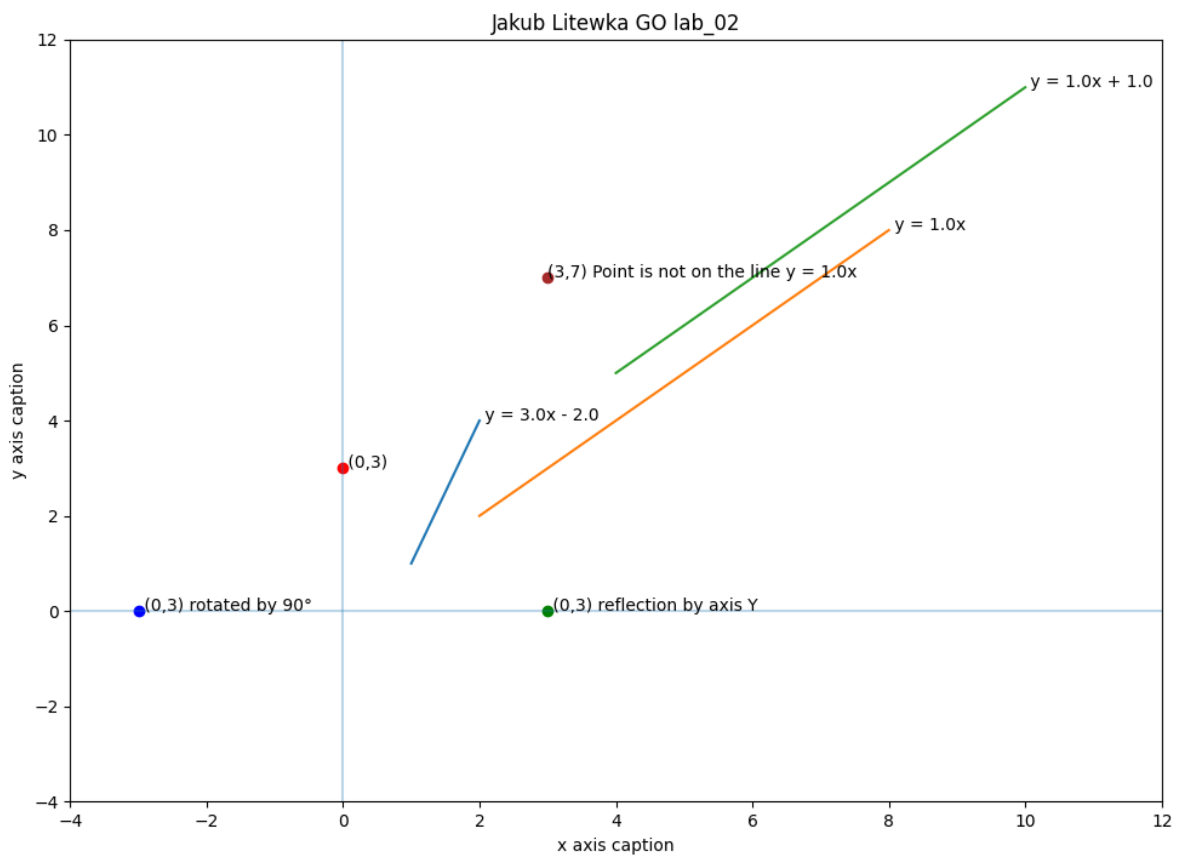
```python
class Point:
    def __init__(self, x: float, y: float) # Point constructor

    def rotate(self, angle: int) # rotate point

    def reflection_OX(self) # reflection about the axis Y

    def reflection_OY(self) # reflection about the axis Y

    def print(self, color: str, text: str | None = '') # draw point
```

```python
class Line:
    def __init__(self, head: Point, tail: Point) # Line constructor

    def f_equation(self) -> str: # return function equation e.g. “y = 2x + 4”

    def print(self) # draw line segment

    def calculateFunction(self) # calculate function a & b

    def vectorTranslation(self, vector: list) # vector translation

    def pointContainsion(self, point: Point) # check if point on line
```

```python
    l1 = Line(Point(1, 1), Point(2, 4)) # create function y = 3x – 2
    l1.print() # draw line segment

    l2 = Line(Point(2, 2), Point(8, 8)) # create function y = x + 1
    l2.print() # draw line segment

    p_rand = Point(random.randint(0, 9), random.randint(0, 9)) # random point
    p_rand.print("brown") # draw point, color = “brown”, text = ‘’
    l2.pointContainsion(p_rand) # check if point on line l2

    l2.vectorTranslation([2, 3]) # translation by vector [2,3]
    l2.print() # draw line segment
    p5 = Point(0, 3) # create point (0,3)
    p5.print("red", '(0,3)')

    p5.rotate(90)
    p5.print("blue", '(0,3) rotated by 90°')

    p5.reflection_OY()
    p5.print("green", '(0,3) reflection by axis Y')
```


Jakub Litewka GO lab_02
y = 1.0x + 1.0
y = 1.0x
(3,7) Point is not on the line y = 1.0x
y = 3.0x - 2.0
(0,3)
(0,3) rotated by 90°
(0,3) reflection by axis Y
x axis caption
y axis caption
−4
−2
0
2
4
6
8
10
12