Homework # 4

01286121 Computer Programming

Software Engineering Program,

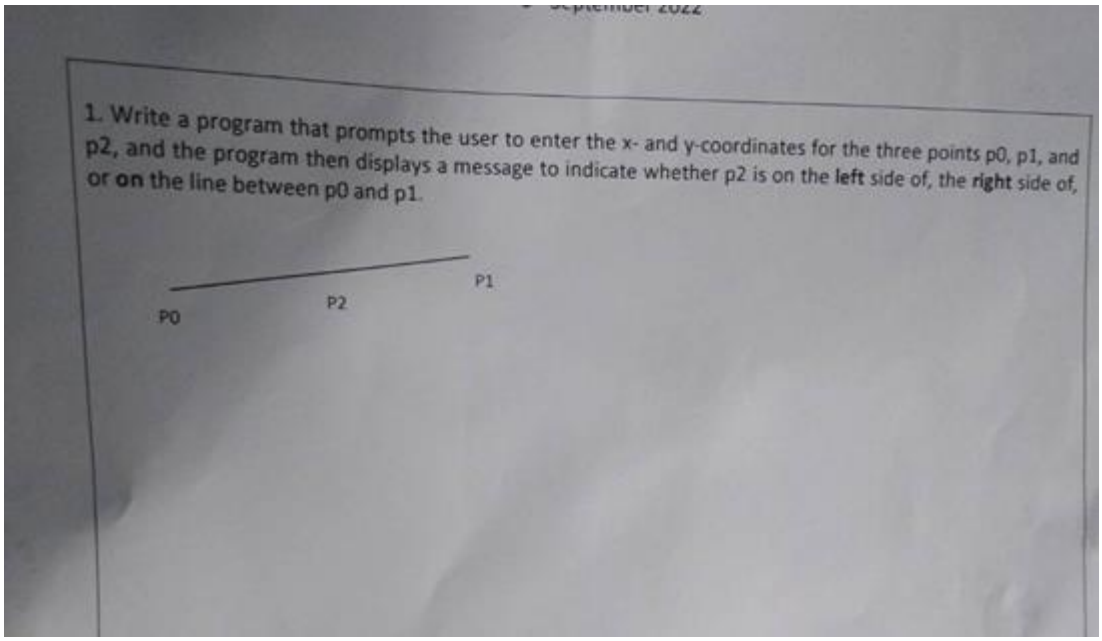
Department of Computer Engineering,

School of Engineering, KMITL

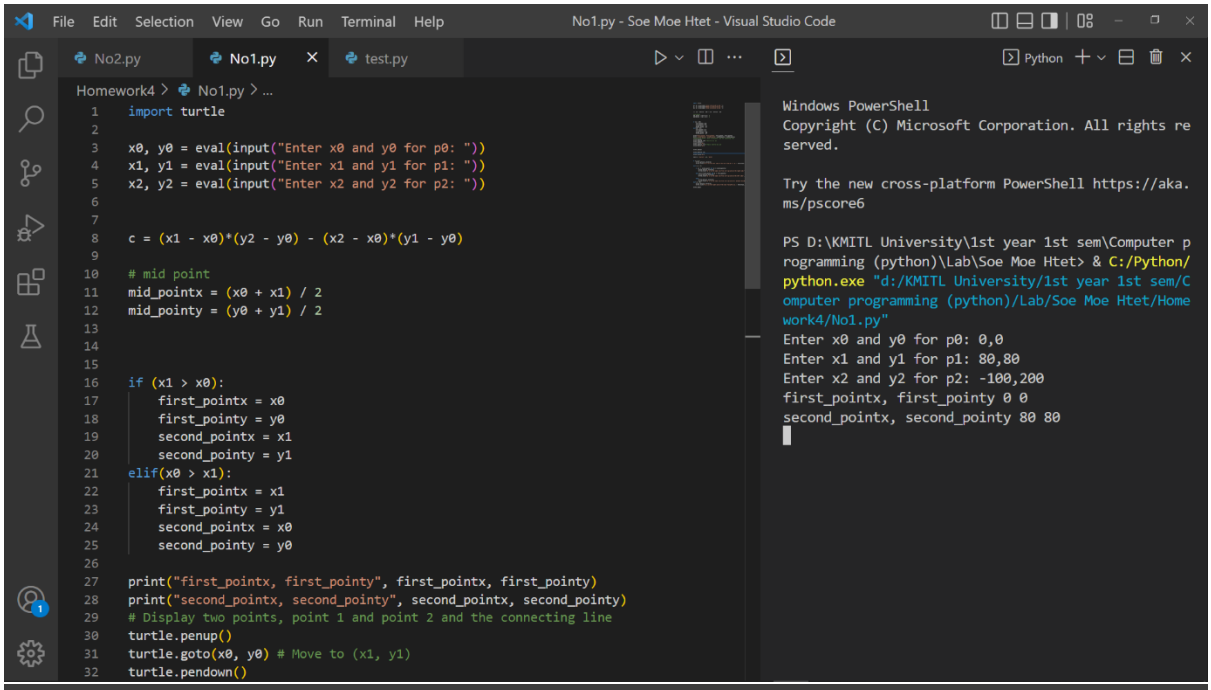
By

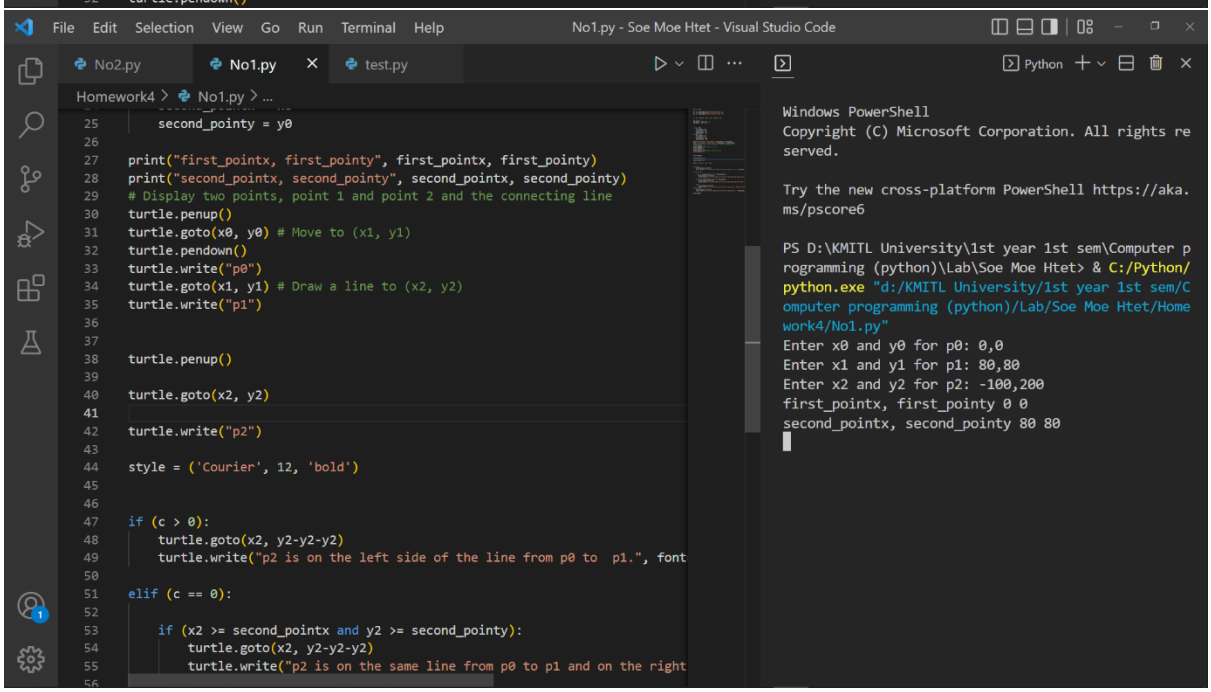
65011693 Soe Moe Htet

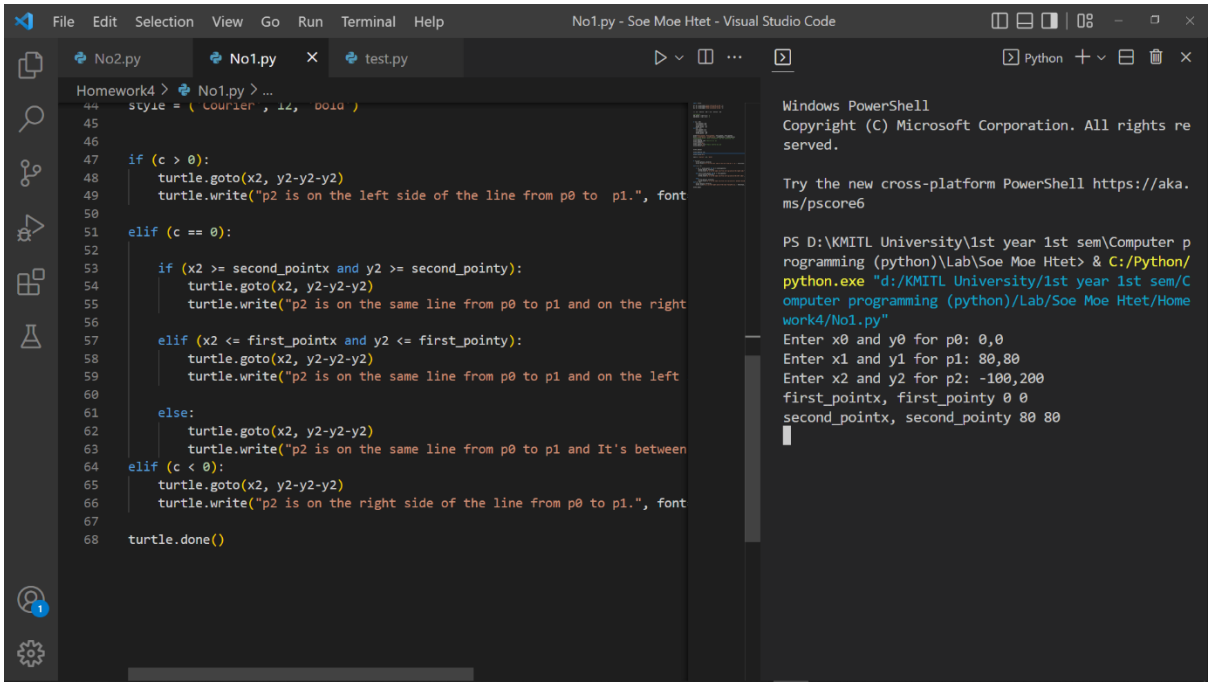
No. 1

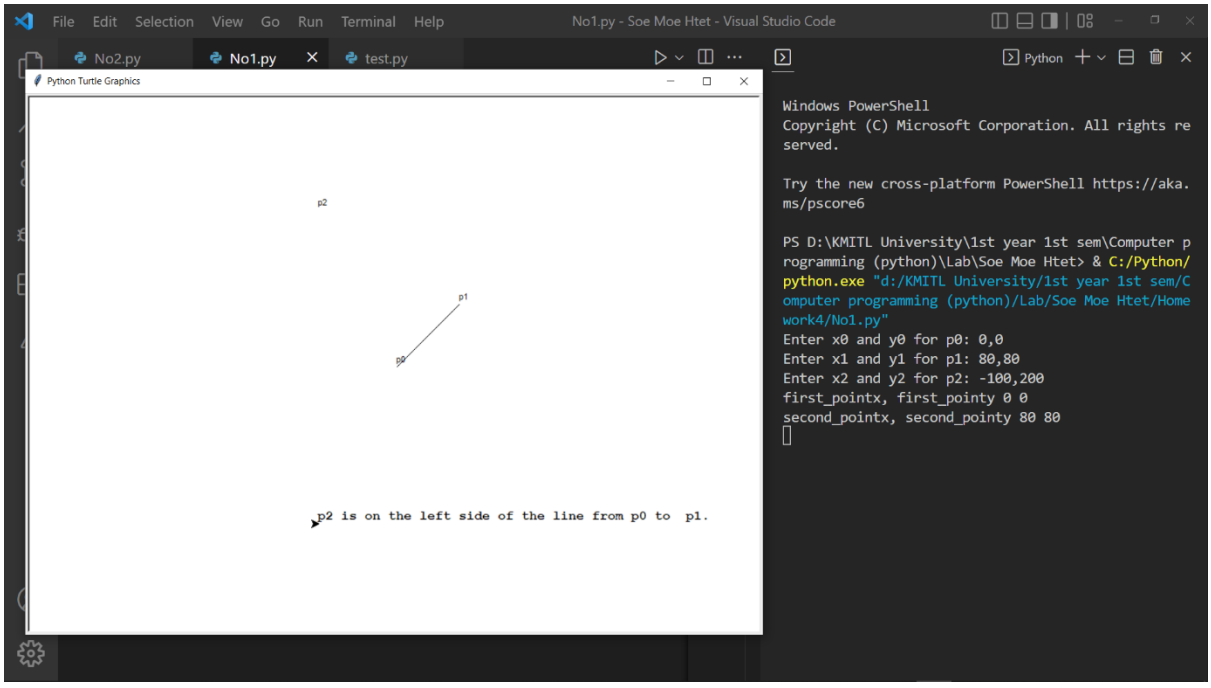


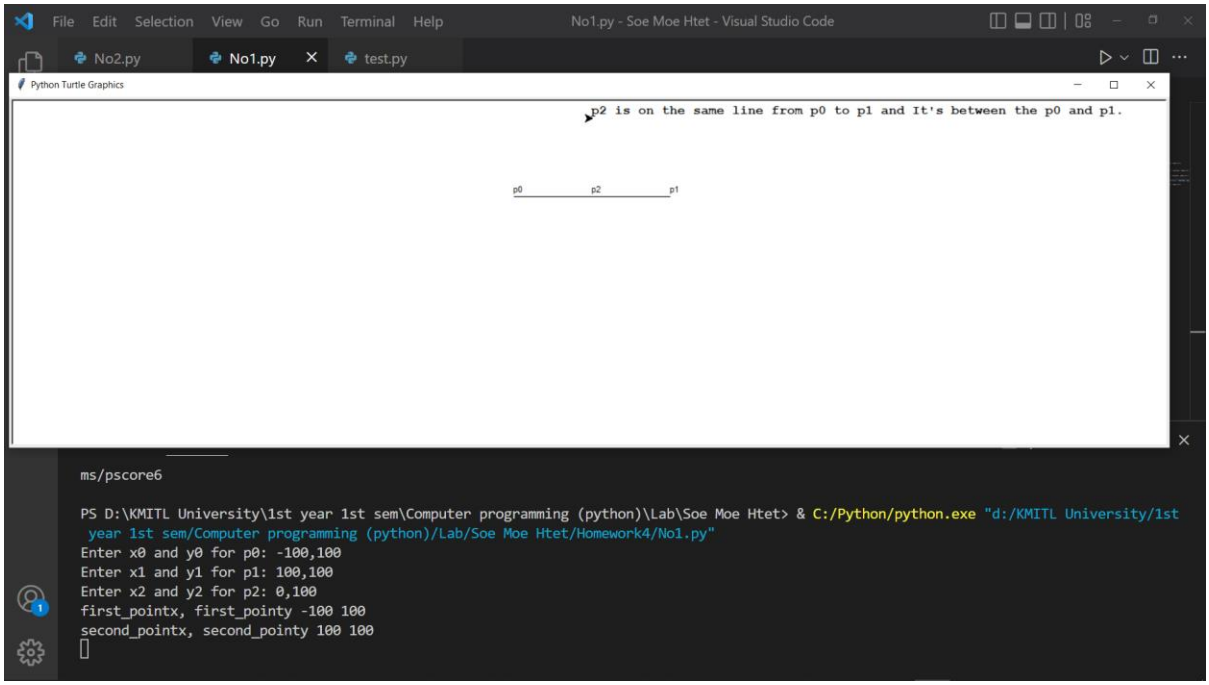
1. Write a program that prompts the user to enter the x- and y-coordinates for the three points p0, p1, and p2, and the program then displays a message to indicate whether p2 is on the **left** side of, the **right** side of, or **on** the line between p0 and p1.

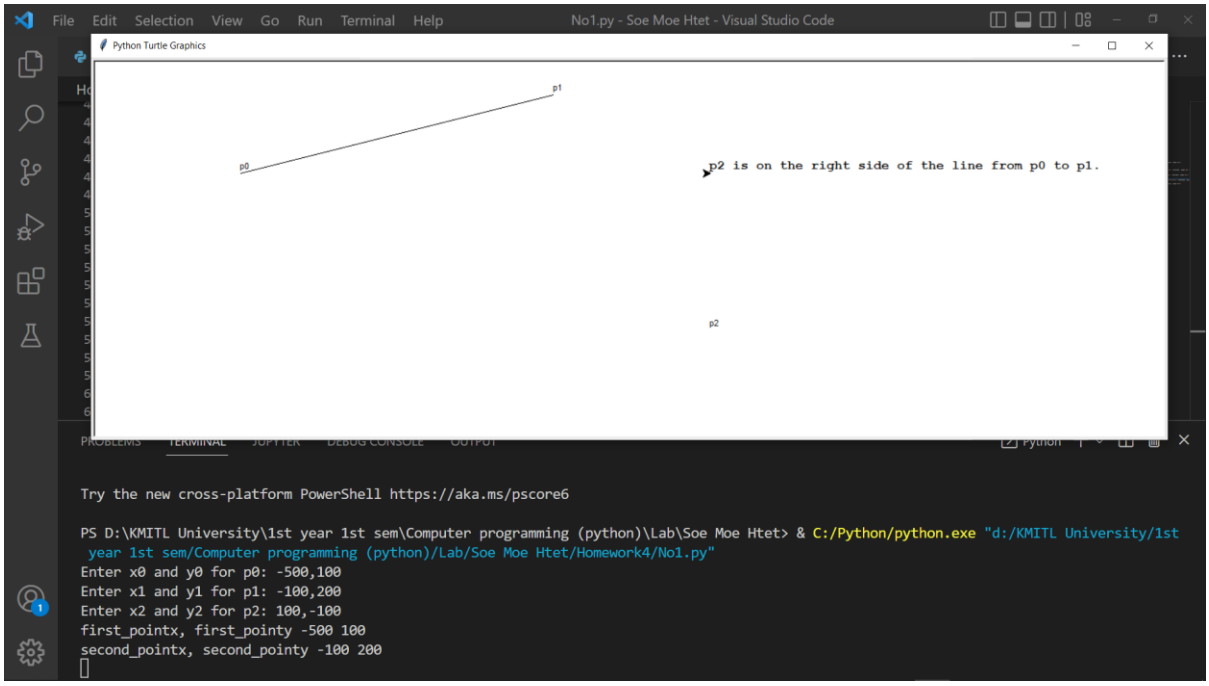

File Edit Selection View Go Run Terminal Help
No1.py - Soe Moe Htet - Visual Studio Code
No2.py
No1.py
test.py
Homework4 > No1.py > ...
import turtle
x0, y0 = eval(input("Enter x0 and y0 for p0: "))
x1, y1 = eval(input("Enter x1 and y1 for p1: "))
x2, y2 = eval(input("Enter x2 and y2 for p2: "))
c = (x1 - x0)*(y2 - y0) - (x2 - x0)*(y1 - y0)
# mid point
mid_pointx = (x0 + x1) / 2
mid_pointy = (y0 + y1) / 2
if (x1 > x0):
first_pointx = x0
first_pointy = y0
second_pointx = x1
second_pointy = y1
elif(x0 > x1):
first_pointx = x1
first_pointy = y1
second_pointx = x0
second_pointy = y0
print("first_pointx, first_pointy", first_pointx, first_pointy)
print("second_pointx, second_pointy", second_pointx, second_pointy)
# Display two points, point 1 and point 2 and the connecting line
turtle.penup()
turtle.goto(x0, y0) # Move to (x1, y1)
turtle.pendown()
Python
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS D:\KMITL University\1st year 1st sem\Computer programming (python)\Lab\Soe Moe Htet> & C:/Python/python.exe "d:/KMITL University/1st year 1st sem/Computer programming (python)/Lab/Soe Moe Htet/Homework4/No1.py"
Enter x0 and y0 for p0: 0,0
Enter x1 and y1 for p1: 80,80
Enter x2 and y2 for p2: -100,200
first_pointx, first_pointy 0 0
second_pointx, second_pointy 80 80


File Edit Selection View Go Run Terminal Help
No1.py - Soe Moe Htet - Visual Studio Code
No2.py
No1.py
test.py
Homework4 > No1.py > ...
second_pointy = y0
print("first_pointx, first_pointy", first_pointx, first_pointy)
print("second_pointx, second_pointy", second_pointx, second_pointy)
# Display two points, point 1 and point 2 and the connecting line
turtle.penup()
turtle.goto(x0, y0) # Move to (x1, y1)
turtle.pendown()
turtle.write("p0")
turtle.goto(x1, y1) # Draw a line to (x2, y2)
turtle.write("p1")
turtle.penup()
turtle.goto(x2, y2)
turtle.write("p2")
style = ('Courier', 12, 'bold')
if (c > 0):
turtle.goto(x2, y2-y2-y2)
turtle.write("p2 is on the left side of the line from p0 to p1.", font
elif (c == 0):
if (x2 >= second_pointx and y2 >= second_pointy):
turtle.goto(x2, y2-y2-y2)
turtle.write("p2 is on the same line from p0 to p1 and on the right
Python
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS D:\KMITL University\1st year 1st sem\Computer programming (python)\Lab\Soe Moe Htet> & C:/Python/python.exe "d:/KMITL University/1st year 1st sem/Computer programming (python)/Lab/Soe Moe Htet/Homework4/No1.py"
Enter x0 and y0 for p0: 0,0
Enter x1 and y1 for p1: 80,80
Enter x2 and y2 for p2: -100,200
first_pointx, first_pointy 0 0
second_pointx, second_pointy 80 80

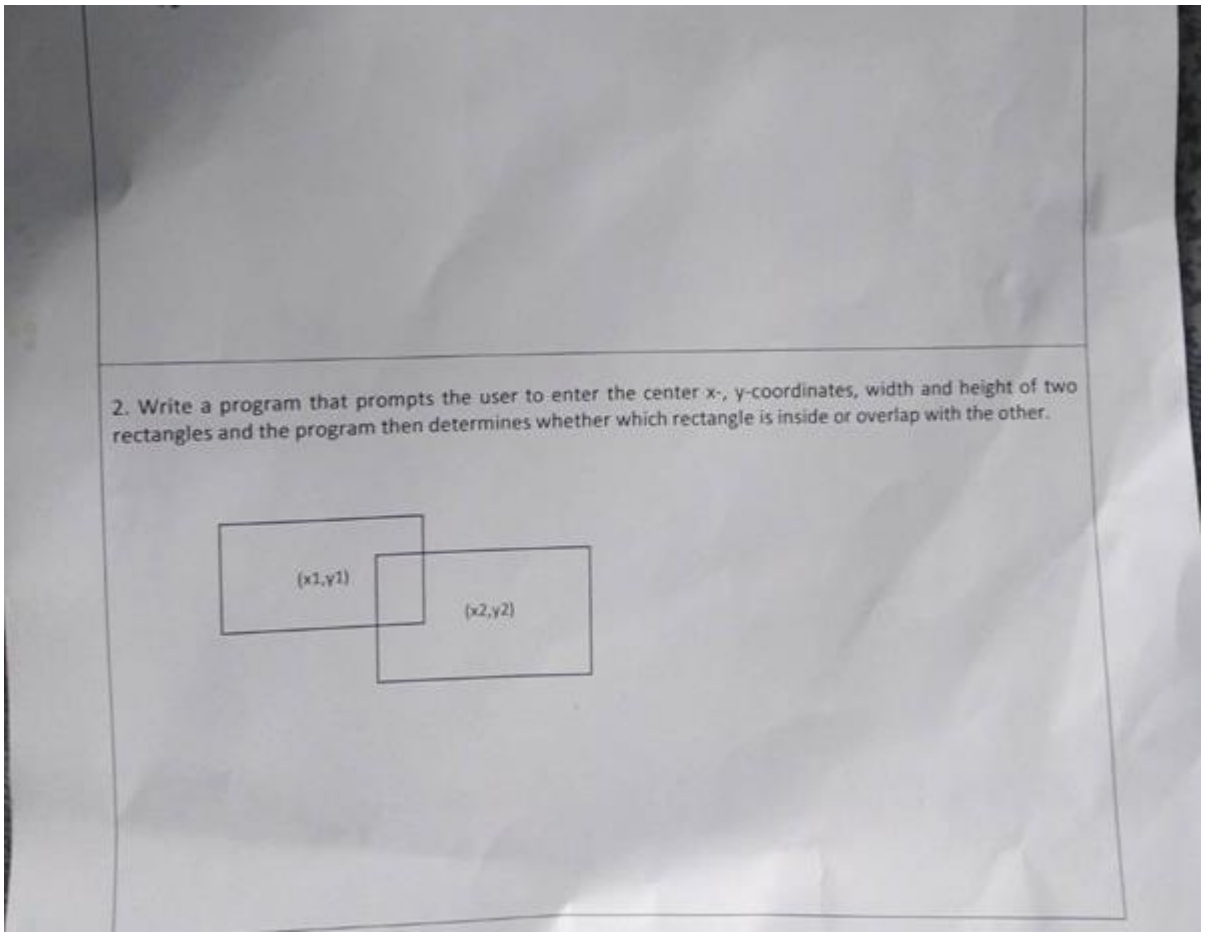

File Edit Selection View Go Run Terminal Help
No1.py - Soe Moe Htet - Visual Studio Code
No2.py
No1.py
test.py
Homework4 > No1.py > ...
style = ('Courier', 12, 'bold')
if (c > 0):
turtle.goto(x2, y2-y2-y2)
turtle.write("p2 is on the left side of the line from p0 to p1.", font
elif (c == 0):
if (x2 >= second_pointx and y2 >= second_pointy):
turtle.goto(x2, y2-y2-y2)
turtle.write("p2 is on the same line from p0 to p1 and on the right
elif (x2 <= first_pointx and y2 <= first_pointy):
turtle.goto(x2, y2-y2-y2)
turtle.write("p2 is on the same line from p0 to p1 and on the left
else:
turtle.goto(x2, y2-y2-y2)
turtle.write("p2 is on the same line from p0 to p1 and It's between
elif (c < 0):
turtle.goto(x2, y2-y2-y2)
turtle.write("p2 is on the right side of the line from p0 to p1.", font
turtle.done()
Python
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS D:\KMITL University\1st year 1st sem\Computer programming (python)\Lab\Soe Moe Htet> & C:/Python/python.exe "d:/KMITL University/1st year 1st sem/Computer programming (python)/Lab/Soe Moe Htet/Homework4/No1.py"
Enter x0 and y0 for p0: 0,0
Enter x1 and y1 for p1: 80,80
Enter x2 and y2 for p2: -100,200
first_pointx, first_pointy 0 0
second_pointx, second_pointy 80 80

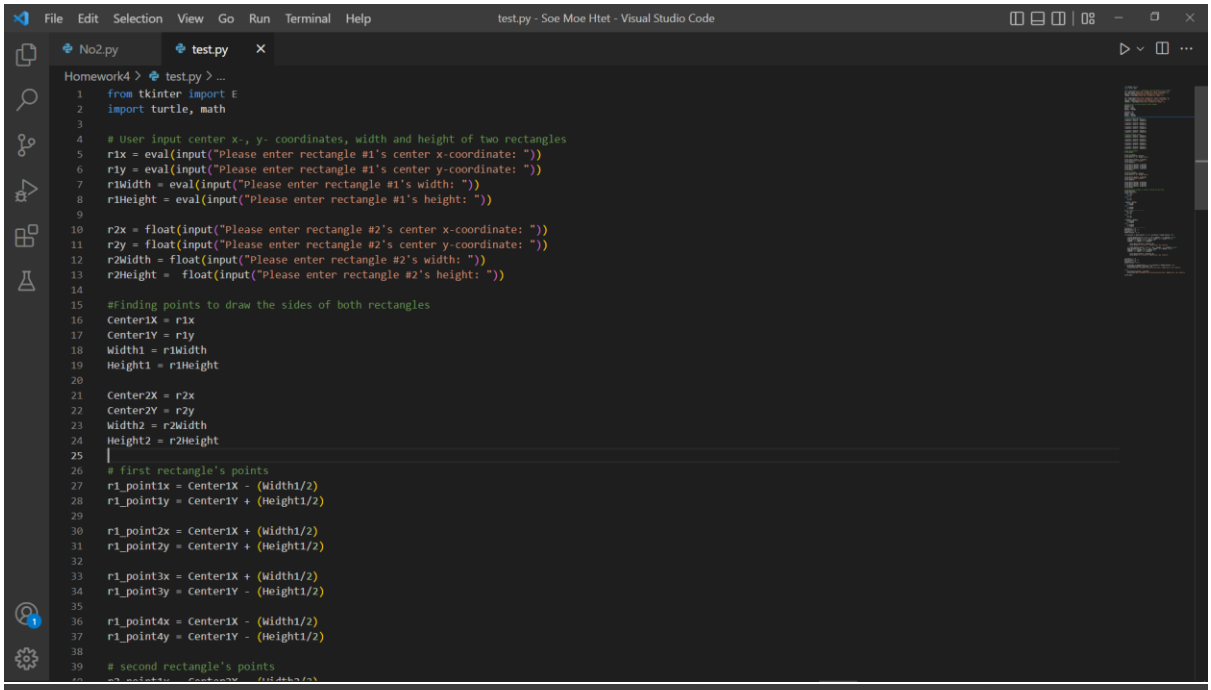

File Edit Selection View Go Run Terminal Help
No1.py - Soe Moe Htet - Visual Studio Code
No2.py
No1.py
test.py
Python Turtle Graphics
p2
p1
p0
p2 is on the left side of the line from p0 to p1.
Python
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS D:\KMITL University\1st year 1st sem\Computer programming (python)\Lab\Soe Moe Htet> & C:/Python/python.exe "d:/KMITL University/1st year 1st sem/Computer programming (python)/Lab/Soe Moe Htet/Homework4/No1.py"
Enter x0 and y0 for p0: 0,0
Enter x1 and y1 for p1: 80,80
Enter x2 and y2 for p2: -100,200
first_pointx, first_pointy 0 0
second_pointx, second_pointy 80 80

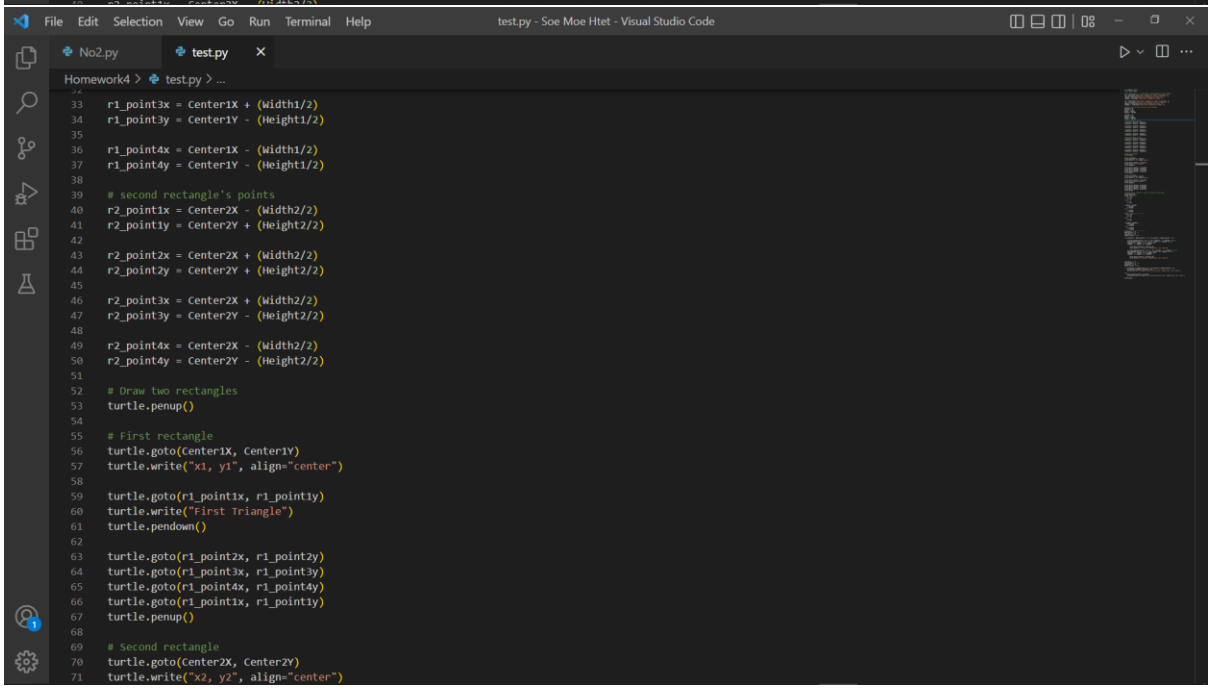

File Edit Selection View Go Run Terminal Help
No1.py - Soe Moe Htet - Visual Studio Code
No2.py
No1.py
test.py
Python Turtle Graphics
p2 is on the same line from p0 to p1 and It's between the p0 and p1.
p0
p2
p1
ms/pscore6
PS D:\KMITL University\1st year 1st sem\Computer programming (python)\Lab\Soe Moe Htet> & C:/Python/python.exe "d:/KMITL University/1st year 1st sem/Computer programming (python)/Lab/Soe Moe Htet/Homework4/No1.py"
Enter x0 and y0 for p0: -100,100
Enter x1 and y1 for p1: 100,100
Enter x2 and y2 for p2: 0,100
first_pointx, first_pointy -100 100
second_pointx, second_pointy 100 100

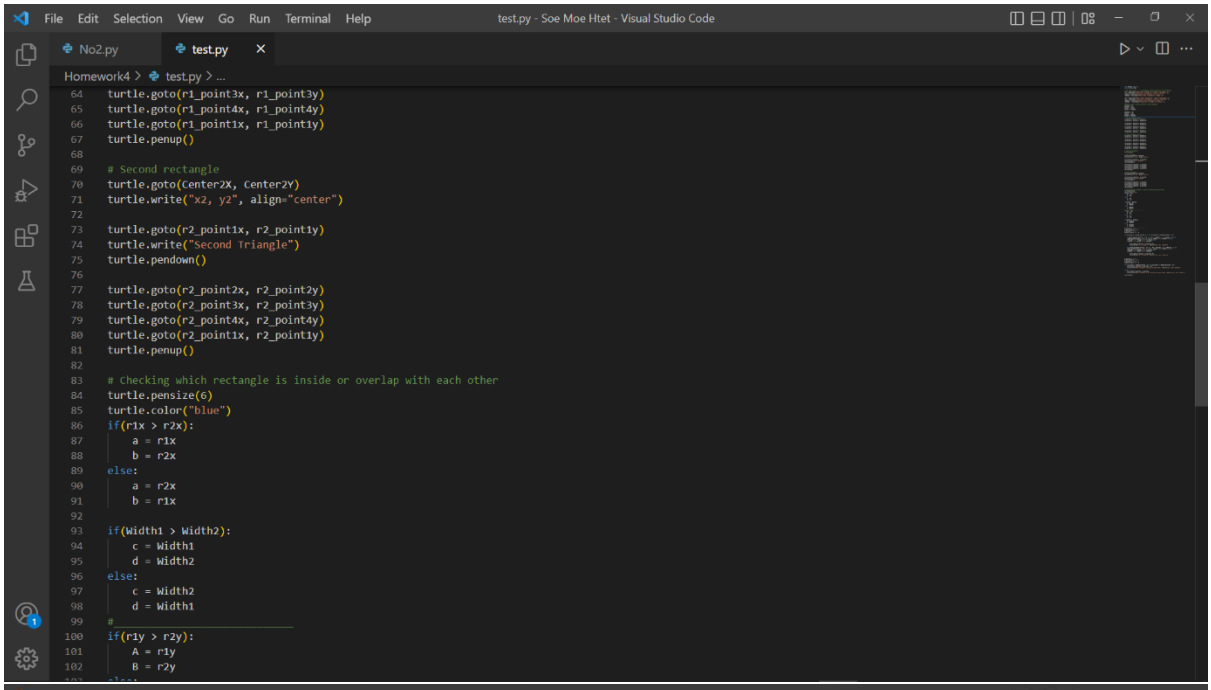

File Edit Selection View Go Run Terminal Help
No1.py - Soe Moe Htet - Visual Studio Code
Python Turtle Graphics
p1
p0
p2 is on the right side of the line from p0 to p1.
p2
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS D:\KMITL University\1st year 1st sem\Computer programming (python)\Lab\Soe Moe Htet> & C:/Python/python.exe "d:/KMITL University/1st year 1st sem/Computer programming (python)/Lab/Soe Moe Htet/Homework4/No1.py"
Enter x0 and y0 for p0: -500,100
Enter x1 and y1 for p1: -100,200
Enter x2 and y2 for p2: 100,-100
first_pointx, first_pointy -500 100
second_pointx, second_pointy -100 200

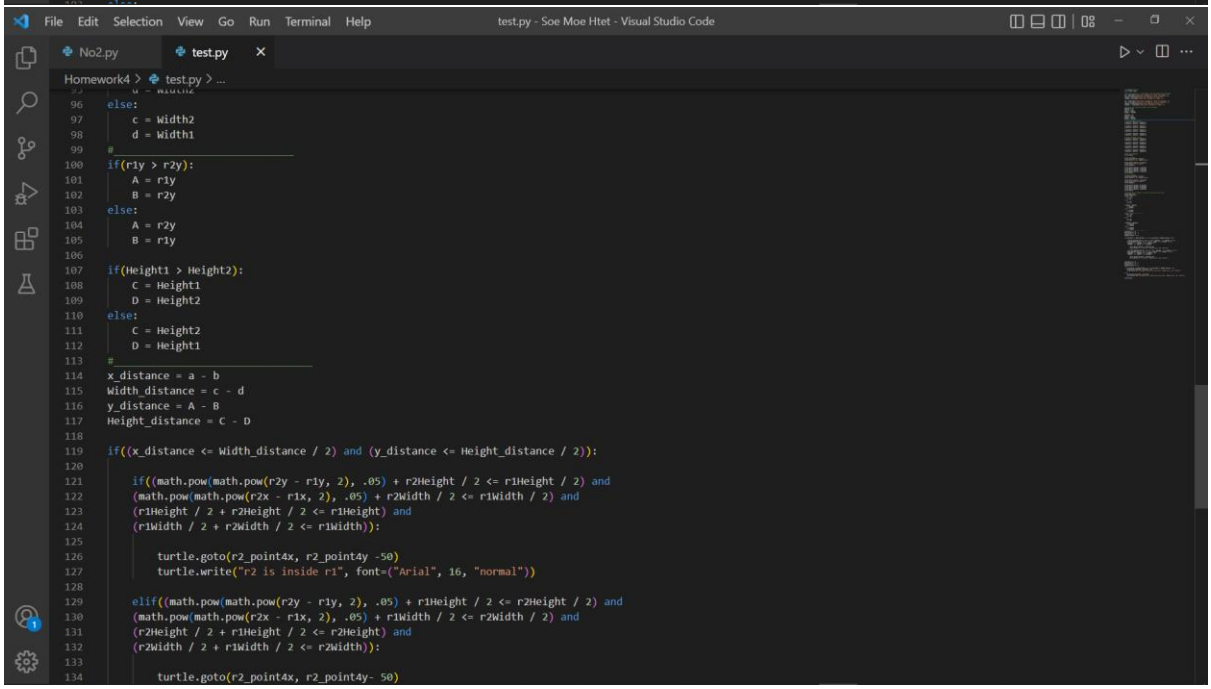
No. 2

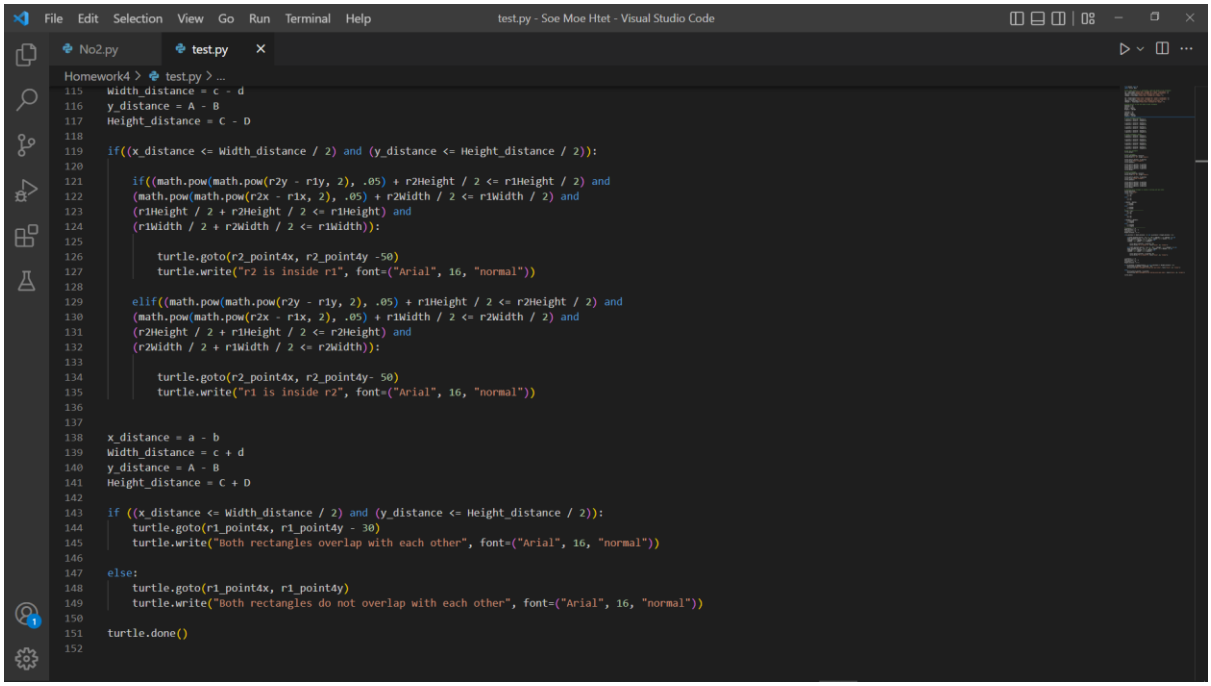

2. Write a program that prompts the user to enter the center x-, y-coordinates, width and height of two rectangles and the program then determines whether which rectangle is inside or overlap with the other.
(x1,y1)
(x2,y2)

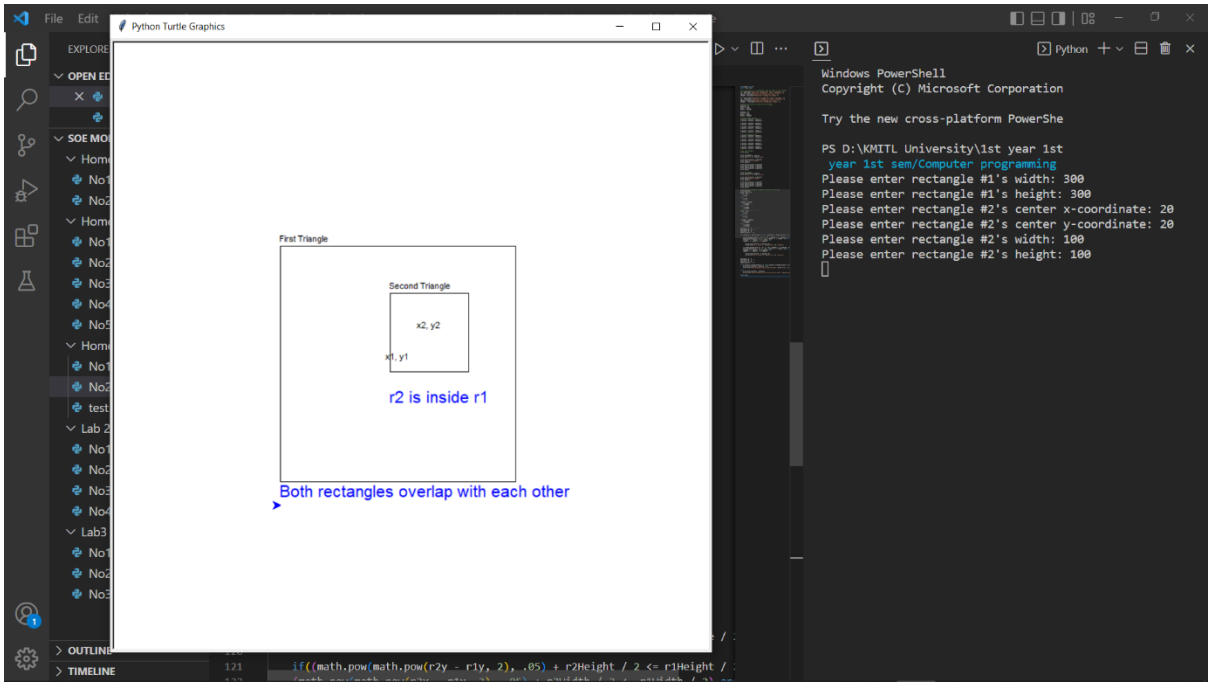

File Edit Selection View Go Run Terminal Help
test.py - Soe Moe Htet - Visual Studio Code
No2.py
test.py
Homework4 > test.py > ...
from tkinter import E
import turtle, math
# User input center x-, y- coordinates, width and height of two rectangles
r1x = eval(input("Please enter rectangle #1's center x-coordinate: "))
r1y = eval(input("Please enter rectangle #1's center y-coordinate: "))
r1Width = eval(input("Please enter rectangle #1's width: "))
r1Height = eval(input("Please enter rectangle #1's height: "))
r2x = float(input("Please enter rectangle #2's center x-coordinate: "))
r2y = float(input("Please enter rectangle #2's center y-coordinate: "))
r2Width = float(input("Please enter rectangle #2's width: "))
r2Height = float(input("Please enter rectangle #2's height: "))
#Finding points to draw the sides of both rectangles
Center1X = r1x
Center1Y = r1y
Width1 = r1Width
Height1 = r1Height
Center2X = r2x
Center2Y = r2y
Width2 = r2Width
Height2 = r2Height
# first rectangle's points
r1_point1x = Center1X - (Width1/2)
r1_point1y = Center1Y + (Height1/2)
r1_point2x = Center1X + (Width1/2)
r1_point2y = Center1Y + (Height1/2)
r1_point3x = Center1X + (Width1/2)
r1_point3y = Center1Y - (Height1/2)
r1_point4x = Center1X - (Width1/2)
r1_point4y = Center1Y - (Height1/2)
# second rectangle's points

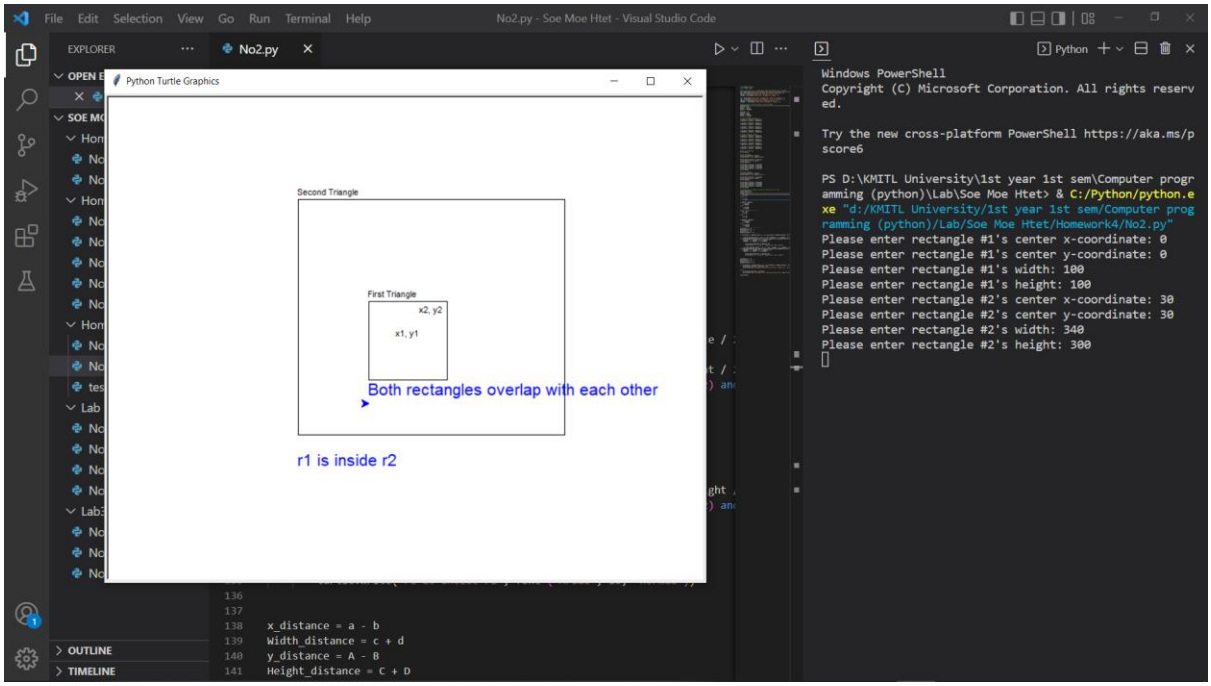

File Edit Selection View Go Run Terminal Help
test.py - Soe Moe Htet - Visual Studio Code
No2.py
test.py
Homework4 > test.py > ...
r1_point3x = Center1X + (Width1/2)
r1_point3y = Center1Y - (Height1/2)
r1_point4x = Center1X - (Width1/2)
r1_point4y = Center1Y - (Height1/2)
# second rectangle's points
r2_point1x = Center2X - (Width2/2)
r2_point1y = Center2Y + (Height2/2)
r2_point2x = Center2X + (Width2/2)
r2_point2y = Center2Y + (Height2/2)
r2_point3x = Center2X + (Width2/2)
r2_point3y = Center2Y - (Height2/2)
r2_point4x = Center2X - (Width2/2)
r2_point4y = Center2Y - (Height2/2)
# Draw two rectangles
turtle.penup()
# First rectangle
turtle.goto(Center1X, Center1Y)
turtle.write("x1, y1", align="center")
turtle.goto(r1_point1x, r1_point1y)
turtle.write("First Triangle")
turtle.pendown()
turtle.goto(r1_point2x, r1_point2y)
turtle.goto(r1_point3x, r1_point3y)
turtle.goto(r1_point4x, r1_point4y)
turtle.goto(r1_point1x, r1_point1y)
turtle.penup()
# Second rectangle
turtle.goto(Center2X, Center2Y)
turtle.write("x2, y2", align="center")

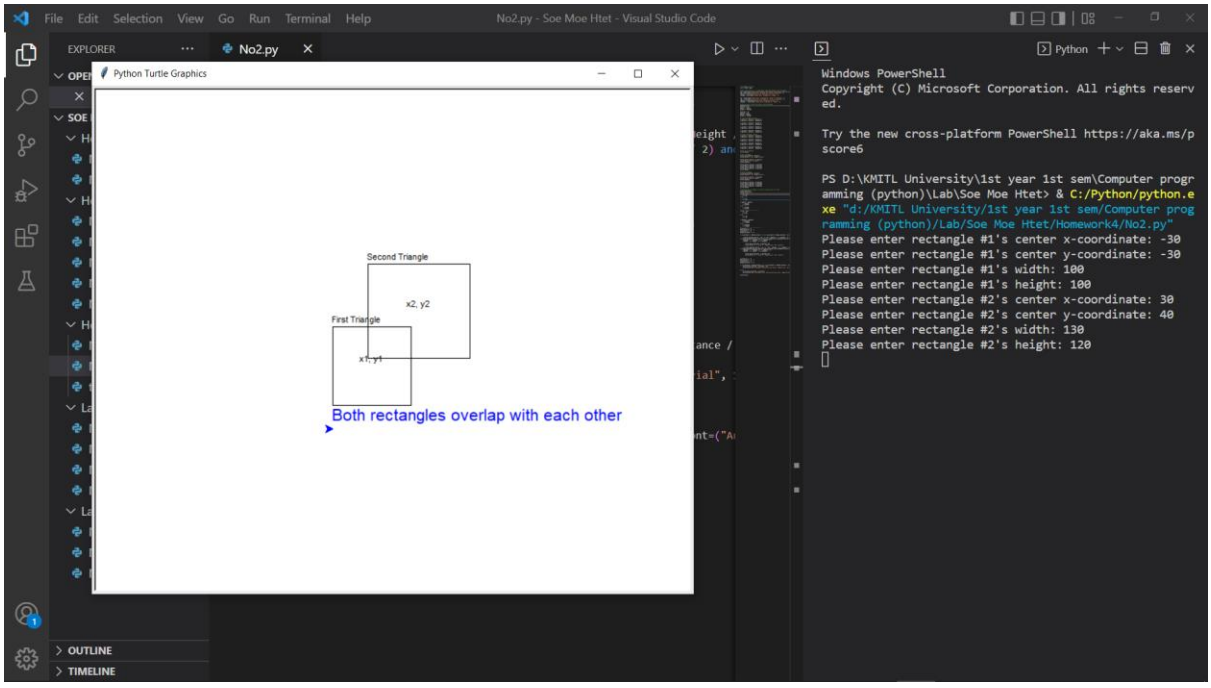

File Edit Selection View Go Run Terminal Help
test.py - Soe Moe Htet - Visual Studio Code
No2.py
test.py
Homework4 > test.py > ...
turtle.goto(r1_point3x, r1_point3y)
turtle.goto(r1_point4x, r1_point4y)
turtle.goto(r1_point1x, r1_point1y)
turtle.penup()
# Second rectangle
turtle.goto(Center2X, Center2Y)
turtle.write("x2, y2", align="center")
turtle.goto(r2_point1x, r2_point1y)
turtle.write("Second Triangle")
turtle.pendown()
turtle.goto(r2_point2x, r2_point2y)
turtle.goto(r2_point3x, r2_point3y)
turtle.goto(r2_point4x, r2_point4y)
turtle.goto(r2_point1x, r2_point1y)
turtle.penup()
# Checking which rectangle is inside or overlap with each other
turtle.pensize(6)
turtle.color("blue")
if(r1x > r2x):
a = r1x
b = r2x
else:
a = r2x
b = r1x
if(Width1 > Width2):
c = Width1
d = Width2
else:
c = Width2
d = Width1
#_____________________
if(r1y > r2y):
A = r1y
B = r2y

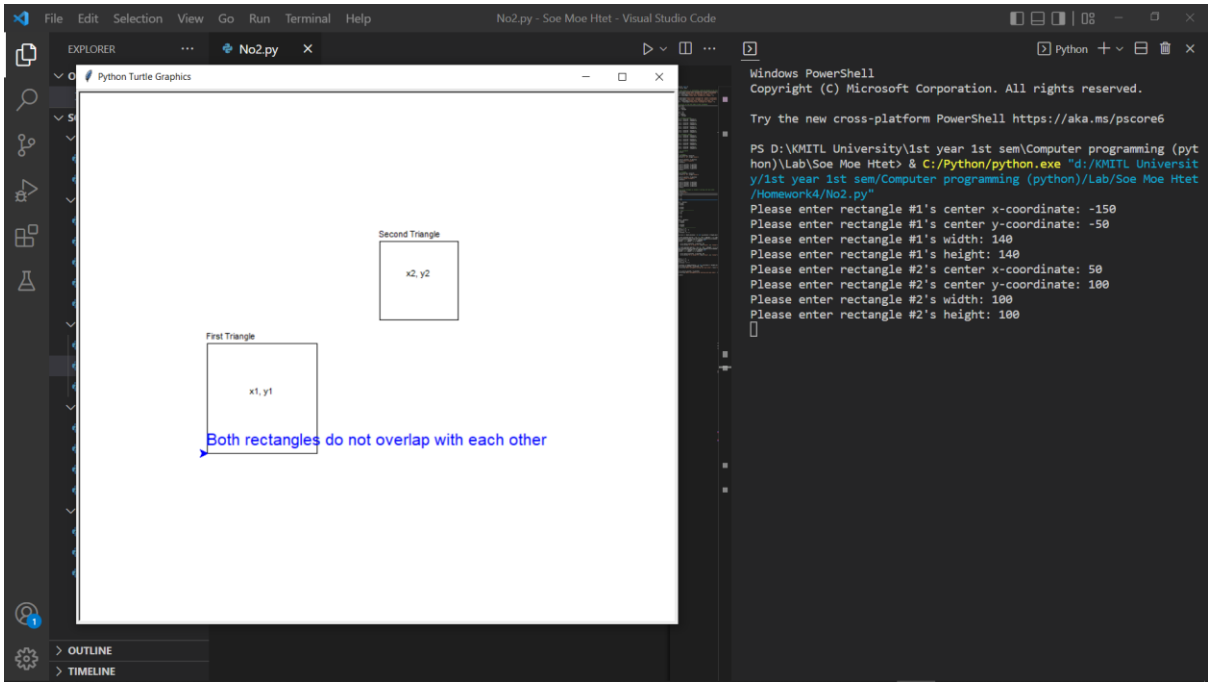

File Edit Selection View Go Run Terminal Help
test.py - Soe Moe Htet - Visual Studio Code
No2.py
test.py
Homework4 > test.py > ...
else:
c = Width2
d = Width1
#_____________________
if(r1y > r2y):
A = r1y
B = r2y
else:
A = r2y
B = r1y
if(Height1 > Height2):
C = Height1
D = Height2
else:
C = Height2
D = Height1
#_____________________
x_distance = a - b
Width_distance = c - d
y_distance = A - B
Height_distance = C - D
if((x_distance <= Width_distance / 2) and (y_distance <= Height_distance / 2)):
if((math.pow(math.pow(r2y - r1y, 2), .05) + r2Height / 2 <= r1Height / 2) and
(math.pow(math.pow(r2x - r1x, 2), .05) + r2Width / 2 <= r1Width / 2) and
(r1Height / 2 + r2Height / 2 <= r1Height) and
(r1Width / 2 + r2Width / 2 <= r1Width)):
turtle.goto(r2_point4x, r2_point4y -50)
turtle.write("r2 is inside r1", font=("Arial", 16, "normal"))
elif((math.pow(math.pow(r2y - r1y, 2), .05) + r1Height / 2 <= r2Height / 2) and
(math.pow(math.pow(r2x - r1x, 2), .05) + r1Width / 2 <= r2Width / 2) and
(r2Height / 2 + r1Height / 2 <= r2Height) and
(r2Width / 2 + r1Width / 2 <= r2Width)):
turtle.goto(r2_point4x, r2_point4y- 50)


File Edit Selection View Go Run Terminal Help
test.py - Soe Moe Htet - Visual Studio Code
No2.py
test.py
Homework4 > test.py > ...
Width_distance = c - d
y_distance = A - B
Height_distance = C - D
if((x_distance <= Width_distance / 2) and (y_distance <= Height_distance / 2)):
if((math.pow(math.pow(r2y - r1y, 2), .05) + r2Height / 2 <= r1Height / 2) and
(math.pow(math.pow(r2x - r1x, 2), .05) + r2Width / 2 <= r1Width / 2) and
(r1Height / 2 + r2Height / 2 <= r1Height) and
(r1Width / 2 + r2Width / 2 <= r1Width)):
turtle.goto(r2_point4x, r2_point4y -50)
turtle.write("r2 is inside r1", font=("Arial", 16, "normal"))
elif((math.pow(math.pow(r2y - r1y, 2), .05) + r1Height / 2 <= r2Height / 2) and
(math.pow(math.pow(r2x - r1x, 2), .05) + r1Width / 2 <= r2Width / 2) and
(r2Height / 2 + r1Height / 2 <= r2Height) and
(r2Width / 2 + r1Width / 2 <= r2Width)):
turtle.goto(r2_point4x, r2_point4y- 50)
turtle.write("r1 is inside r2", font=("Arial", 16, "normal"))
x_distance = a - b
Width_distance = c + d
y_distance = A - B
Height_distance = C + D
if ((x_distance <= Width_distance / 2) and (y_distance <= Height_distance / 2)):
turtle.goto(r1_point4x, r1_point4y - 30)
turtle.write("Both rectangles overlap with each other", font=("Arial", 16, "normal"))
else:
turtle.goto(r1_point4x, r1_point4y)
turtle.write("Both rectangles do not overlap with each other", font=("Arial", 16, "normal"))
turtle.done()


Python Turtle Graphics
First Triangle
Second Triangle
x2, y2
x1, y1
r2 is inside r1
Both rectangles overlap with each other
Windows PowerShell
Copyright (C) Microsoft Corporation
Try the new cross-platform PowerShe
PS D:\KMITL University\1st year 1st
year 1st sem/Computer programming
Please enter rectangle #1's width: 300
Please enter rectangle #1's height: 300
Please enter rectangle #2's center x-coordinate: 20
Please enter rectangle #2's center y-coordinate: 20
Please enter rectangle #2's width: 100
Please enter rectangle #2's height: 100


File Edit Selection View Go Run Terminal Help
No2.py - Soe Moe Htet - Visual Studio Code
EXPLORER
No2.py
Python Turtle Graphics
Second Triangle
First Triangle
x2, y2
x1, y1
Both rectangles overlap with each other
r1 is inside r2
x_distance = a - b
Width_distance = c + d
y_distance = A - B
Height_distance = C + D
OUTLINE
TIMELINE
Python
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS D:\KMITL University\1st year 1st sem\Computer programming (python)\Lab\Soe Moe Htet> & C:/Python/python.exe "d:/KMITL University/1st year 1st sem/Computer programming (python)/Lab/Soe Moe Htet/Homework4/No2.py"
Please enter rectangle #1's center x-coordinate: 0
Please enter rectangle #1's center y-coordinate: 0
Please enter rectangle #1's width: 100
Please enter rectangle #1's height: 100
Please enter rectangle #2's center x-coordinate: 30
Please enter rectangle #2's center y-coordinate: 30
Please enter rectangle #2's width: 340
Please enter rectangle #2's height: 300


File Edit Selection View Go Run Terminal Help
No2.py - Soe Moe Htet - Visual Studio Code
EXPLORER
No2.py
Python Turtle Graphics
Second Triangle
First Triangle
x2, y2
x1, y1
Both rectangles overlap with each other
OUTLINE
TIMELINE
Python
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS D:\KMITL University\1st year 1st sem\Computer programming (python)\Lab\Soe Moe Htet> & C:/Python/python.exe "d:/KMITL University/1st year 1st sem/Computer programming (python)/Lab/Soe Moe Htet/Homework4/No2.py"
Please enter rectangle #1's center x-coordinate: -30
Please enter rectangle #1's center y-coordinate: -30
Please enter rectangle #1's width: 100
Please enter rectangle #1's height: 100
Please enter rectangle #2's center x-coordinate: 30
Please enter rectangle #2's center y-coordinate: 40
Please enter rectangle #2's width: 130
Please enter rectangle #2's height: 120


File Edit Selection View Go Run Terminal Help
No2.py - Soe Moe Htet - Visual Studio Code
EXPLORER
No2.py
Python Turtle Graphics
Second Triangle
x2, y2
First Triangle
x1, y1
Both rectangles do not overlap with each other
OUTLINE
TIMELINE
Python
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS D:\KMITL University\1st year 1st sem\Computer programming (python)\Lab\Soe Moe Htet> & C:/Python/python.exe "d:/KMITL University/1st year 1st sem/Computer programming (python)/Lab/Soe Moe Htet/Homework4/No2.py"
Please enter rectangle #1's center x-coordinate: -150
Please enter rectangle #1's center y-coordinate: -50
Please enter rectangle #1's width: 140
Please enter rectangle #1's height: 140
Please enter rectangle #2's center x-coordinate: 50
Please enter rectangle #2's center y-coordinate: 100
Please enter rectangle #2's width: 100
Please enter rectangle #2's height: 100