Лабораторна робота №4

Типи і класи типів

студента групи Анд-31

Горелика Андрія

Варіант 1

Мета роботи:

Ознайомитись з системою типів та класів типів. Набути досвіду визначення нових типів та класів типів і їх використання.

Завдання 1

Фiгури на площинi. Використовуються такi фiгури, як коло (центр та радiус), прямокутник (координати лiвої верхньої та правої нижньої точок), трикутник (координати вершин) та мiтка — label (координати лiвої нижньої точки, шрифт та рядок). Обчислити площi фiгури (для мiтки — через розмiр одного символу та їх кiлькiсть)

Побудована функція для обчислення площі круга за вказаним центром і радіусом

data Circle = Circle Float Float Float

x :: Circle -> Float

x (Circle x1p \_ \_ ) = x1p

y :: Circle -> Float

y (Circle \_ x2p \_ ) = x2p

radius :: Circle -> Float

radius (Circle \_ \_ r ) = r

circleList :: [Circle]

circleList = [Circle 0 0 3,

              Circle 0 0 4,

              Circle 1 1 2]

squareCircle :: [Circle] -> [Float]

squareCircle [] = []

squareCircle (x : xs) = 3.1415 \* radius x \* radius x : squareCircle xs

Вхідні дані:

squareCircle circleList

Вихідні дані:

[28.273499,50.264,12.566]

Побудована функція для обчислення площі прямокутника за вказаними протилежними точками прямокутника

data Rectangle = Rectangle Float Float Float Float

x1 :: Rectangle -> Float

x1 (Rectangle x1p \_ \_ \_ ) = x1p

y1 :: Rectangle -> Float

y1 (Rectangle \_ y1p \_ \_ ) = y1p

x2 :: Rectangle -> Float

x2 (Rectangle \_ \_ x2p \_ ) = x2p

y2 :: Rectangle -> Float

y2 (Rectangle \_ \_ \_ y2p ) = y2p

rectangleList :: [Rectangle]

rectangleList = [

    Rectangle 0 0 1 1,

    Rectangle 2 2 5 5,

    Rectangle 0 (-1) 1 0 ]

squareRectangle :: [Rectangle] -> [Float]

squareRectangle [] = []

squareRectangle (x : xs) = abs(x2 x - x1 x) \* abs(y2 x - y1 x) : squareRectangle xs

Вхідні дані:

squareTriangle triangleList

Вихідні дані:

[6.0,37.94733,103.92305]

Побудована функція для обчислення площі трикутника за вказаними сторонами трикутника

data Triangle = Triangle Float Float Float

x\_1 :: Triangle -> Float

x\_1 (Triangle x1p \_ \_ ) = x1p

x\_2 :: Triangle -> Float

x\_2 (Triangle \_ x2p \_ ) = x2p

x\_3 :: Triangle -> Float

x\_3 (Triangle \_ \_ x3p ) = x3p

triangleList :: [Triangle]

triangleList = [Triangle 4 3 5,

            Triangle 11 7 12,

            Triangle 12 20 28]

squareTriangle :: [Triangle] -> [Float]

squareTriangle [] = []

squareTriangle (x : xs) = sqrt((x\_1 x + x\_2 x + x\_3 x)\*0.5\*((x\_1 x + x\_2 x + x\_3 x)\*0.5 - x\_1 x)\*((x\_1 x + x\_2 x + x\_3 x)\*0.5 - x\_2 x)\*((x\_1 x + x\_2 x + x\_3 x)\*0.5 - x\_3 x)) : squareTriangle xs

Вхідні дані:

squareRectangle rectangleList

Вихідні дані:

[1.0,9.0,1.0]

Висновок:

Під час даної лабораторної роботи ознайомився з класами мови Haskell, також було створено клас, який ідентифікує фігури та визначає їхні площі