**Задачи**

Дадени са следните протоколи (интерфейси):

Prinrable

protocol Visual {

var text: String { get }

func render()

}

VisualComponent

protocol VisualComponent {

//минимално покриващ правоъгълник

var boundingBox: Rect { get }

var parent: VisualComponent? { get }

func draw()

}

VisualGroup

protocol VisualGroup: VisualComponent {

//броят деца

var numChildren: Int { get }

//списък от всички деца

var children: [VisualComponent] { get }

//добавяне на дете

func add(child: VisualComponent)

//премахване на дете

func remove(child: VisualComponent)

//премахване на дете от съответния индекс - 0 базиран

func removeChild(at: Int)

}

1. Да се имплементират следните класове (или структури, *по избор*):
   * Triangle: VisualComponent, Visual
     + коструктор Trinagle(a: Point, b: Point, c: Point)
   * Rectangle: VisualComponent, Visual
     + коструктор Rectangle(x: Int, y: Int, width: Int, height: Int)
   * Circle: VisualComponent, Visual
     + конструктор Circle(x: Int, y:Int, r: Double)
   * Path: VisualComponent, Visual
     + конструктор Path(points: [Point])
   * HGroup: VisualGroup, Visual
     + конструктор HGroup()
   * VGroup: VisaulGroup, Visual
     + конструктор VGroup()
2. Да се имплементира функция, която определя дълбочината на йерархия от VisualComponent func depth(root: VisualComponent?) -> Int

Пример:

Ако

root =

HGroup

VGroup

Circle

HGroup

Square

Circle

Circle

Circle

тогава

depth(root: roоt) трябва да се оцени до 4

1. Да се имплемнтира функция, която определя броят елементите от даден "клас" / тип в йерархия от VisualComponent. func count(root: VisualComponent?, type: VisualComponent) -> Int

Пример:

Ако

c = Cicle

root =

HGroup

VGroup

Circle

Square

Circle

Circle

Circle

тогава

count(root: roоt, type: c) трябва да се оцени до 4

1. Да се напише функция, която намира най-малкия покриващ правоъгълник на VisualComponent.

func cover(root: VisualComponent?) -> Rect

Пример:

Ако

root =

HGroup

VGroup

Circle (x:0, y:0, r:1)

тогава

cover(root: roоt) трябва да оцени до Rect(x: -1, y: 1, width: 2, height: 2)

1. Да се напише функция, която прочита от символен низ йерархия, и връща нейния модел, използвайки горните класове/структури.

func read(from: String?) -> VisualComponent?

HGroup[VGroup[Circle(0,0,1)],HGroup[Path[(1,1),(2,2)], Circle(1,1,1)]]

Резултатът от входните данни по-горе трябва да е **един** обект от тип HGroup, който има две деца. Първото VGroup(с един Circle), а втортото отново HGroup с Path от 2 точки и Circle.

**Помощни структури**

struct Point {

var x: Double

var y: Double

}

struct Rect {

//top-left

var top:Point

var width: Double

var height: Double

init(x: Double, y: Double, width: Double, height: Double) {

top = Point(x: x, y: y)

self.width = width

self.height = height

}

}