

ЛЕКЦІЯ №**07****ДАТА ПЕРЕГЛЯДУ:****20/05/2018****ПЕРЕВІРЯВ:****Горбушко К****СТУДЕНТ:****Володимир Русиник****ГІЛКА:****Lesson-7****ОЦІНКА:****5 / 10**

Залито pdf завдання - не потрібно

ЗАВДАННЯ 1

ОК

ЗАВДАННЯ 2

відстуня функція для сортування з вибраним типом сортування використано функцію `swarAt` - за умовою завдання - «Реалізація функції не повинна викликати жодну вбудовану функцію»

ЗАВДАННЯ 3

ви буквально підійшли до завдання, а що якби в алфавіті було б 10000 знаків - наприклад китайський - там 3500 символів - описували б всі символи?

Спробуйте реалізувати за допомогою `CharacterSet`

Завжди коли бачите завдання уявіть найпростіший і найгірший варіанти.

Які обмеження ви бачитимете? Який підхід допоможе їх обійти?

Співставте з вимогами до завдання

ЗАВДАННЯ 4

Reduce хороший варіант

Спробуйте описати без вбудованого функціоналу одним або кількома циклами

ЗАВДАННЯ 5

Остача від ділення

```
func division(_ a: Int, _ b: Int) -> Int {  
    return a % b  
}
```

Піднесення до степеня

```
func pow(_ a: Int, _ b: Int) -> Int {  
    var powerResult : Int = 1  
    for _ in 1...b {  
        power *= a  
    }  
    return powerResult  
}
```

Це мав бути консольний додаток - адже вибір як введення від користувача в playground недоступно

Ви мали запитати користувача на введення кількох символів і потім зчитати їх і виконати послідовно всі операції

зчитати з консолі - `readLine()`

вивести - `print()`

ваша ідея побудови коду - ок, але бажано спробувати в консолі

ЗАВДАННЯ 6

Скину варіант рішення від одного з студентів вашої групи

```
func maxCircleWithPoint(_ circleArray:[(x:Double, y:Double, rad:Double)],
_ point:(p1: Double, p2: Double)) -> (Double, Double, Double){
    var maxCircle : (x: Double, y: Double, rad: Double) = (0.0,0.0,0.0)
    // сортуємо від максимального радіуса до найменшого
    let circleArray = circleArray.sorted(by: {$0.rad > $1.rad})

    for t in circleArray {
        // належність точки колу визначається теоремою Піфагора:
        // (p1-center_x)^2 + (p2 - center_y)^2 < radius^2
        let d = pow((point.p1-t.x),2) + pow((point.p2-t.y),2)

        // якщо точка в колі або на колі записуємо з змінну
maxCircle і виходимо
        // інакше, перевіряємо наступні менші радіуси
        if d <= pow(t.rad,2) {
            maxCircle = (t.x,t.y,t.rad)
            print("point: \(point) inside max circle: \(maxCircle)")
            break
        }
        else {
            print("point: \(point) is outside a circle: \(t)")
        }
    }
    return maxCircle
}
```