Министерство образования

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

Отчет по лабораторной работе №3

«Использование языка программирования Swift: структуры и классы, методы»

Выполнил: Сверчков Е.А.

Студент группы 310902

Проверил: Усенко Ф.В.

Минск 2024

Индивидуальное задание №11: при инициализации класс принимает целое нечетное число – сторону квадрата, в который вписана снежинка. Методы: thaw() – таять, при этом на каждом шаге пропадают крайние звездочки со всех сторон; параметр показывает, сколько шагов прошло; freeze(n) – намораживаться, при этом сторона квадрата, в который вписана снежинка, увеличивается на 2 \* n, одновременно добавляются звездочки в нужных местах, чтобы правило соблюдалось; thicken() – утолщаться, ко всем линиям звездочек с двух сторон добавляются параллельные (если перед этим снежинка таяла, то теперь звездочки восстанавливаются); show() – показывать (рисуется снежинка в виде квадратной матрицы со звездочками и дефисами в пустых местах).

Листинг кода программы:

Файл Snowflake.swift:

class Snowflake {

private var side: Int

private var matrix: [[Character]]

private var thawSteps: Int = 0

init(side: Int) {

if side % 2 != 1 || side < 1 {

fatalError("Side must be an positive odd number.")

}

self.side = side

self.matrix = Array(repeating: Array(repeating: "-", count: side), count: side)

self.createSnowflake()

}

private func createSnowflake() {

let center = side / 2

for i in 0..<side {

matrix[i][center] = "\*"

}

for i in 0..<side {

matrix[center][i] = "\*"

}

for i in 0..<side {

for j in 0..<side {

if i == j {

matrix[i][j] = "\*"

}

if i + j == side - 1 {

matrix[i][j] = "\*"

}

}

}

}

func thaw() {

thawSteps += 1

for i in 0..<side {

for j in 0..<side {

if i == thawSteps-1 {

matrix[i][j] = "-"

}

if j == thawSteps-1 {

matrix[i][j] = "-"

}

if i == side - thawSteps {

matrix[i][j] = "-"

}

if j == side - thawSteps {

matrix[i][j] = "-"

}

}

}

}

func freeze(n: Int) {

side += 2 \* n

var newMatrix: [[Character]] = Array(repeating: Array(repeating: "-", count: side), count: side)

for i in 0..<matrix.count {

for j in 0..<matrix[i].count {

newMatrix[i+n][j+n] = matrix[i][j]

}

}

matrix = newMatrix

createSnowflake()

}

func thicken() {

if (thawSteps != 0){

createSnowflake();

thawSteps = 0;

return;

}

var k = 0;

while k < side && matrix[0][k] != "-" {

k+=1;

}

var n = side-k

for i in 0..<n {

matrix[i][k] = "\*"

matrix[side-1-i][k] = "\*"

matrix[k][i] = "\*"

matrix[side-1-k][i] = "\*"

k += 1

}

let centre = side / 2

k = side / 2

while k < side && matrix[k][0] != "-" {

k+=1;

}

for i in 0..<side {

matrix[k][i] = "\*"

matrix[2\*centre - k][i] = "\*"

matrix[i][k] = "\*"

matrix[i][2\*centre - k] = "\*"

}

}

func show() {

for row in matrix {

var str: String = ""

for element in row {

str += String(element)

str += " "

}

print(str)

}

}

}

Файл main.swift:

var snowflake = Snowflake(side: 13)

snowflake.show()

snowflake.thaw()

print("\n------------------\n")

snowflake.show()

snowflake.thaw()

print("\n------------------\n")

snowflake.show()

snowflake.thicken()

print("\n------------------\n")

snowflake.show()

snowflake.thicken()

print("\n------------------\n")

snowflake.show()

snowflake.thaw()

print("\n------------------\n")

snowflake.show()

snowflake.thaw()

print("\n------------------\n")

snowflake.show()

snowflake.thaw()

print("\n------------------\n")

snowflake.show()

Результат работы программы:

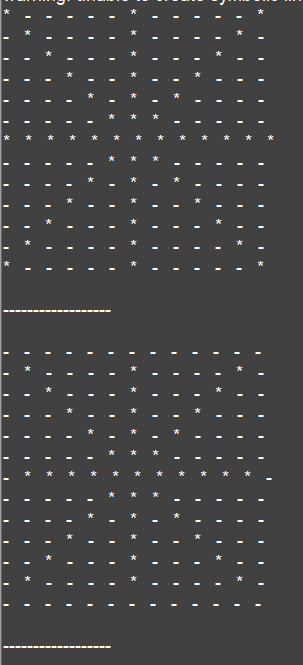


Рисунок 1 – Заполнение и таяние снежинки

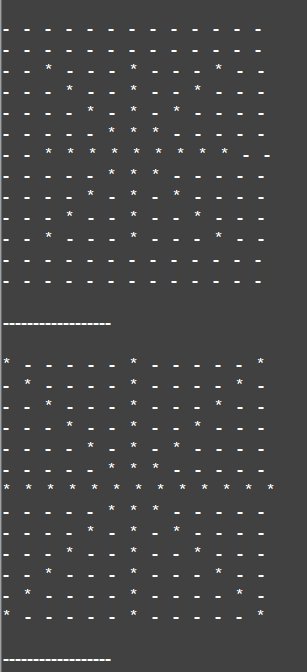
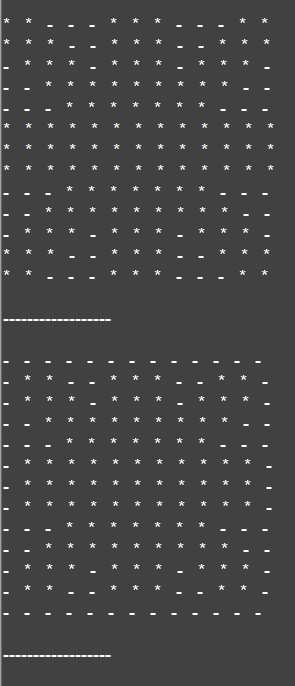


Рисунок 2, 3 –Таяние, две заморозки и ещё раз таяние снежинки

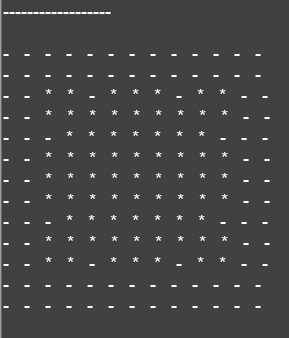
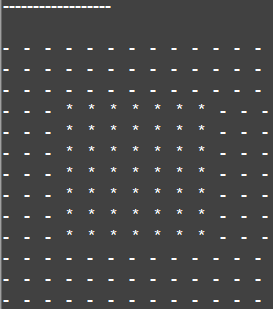


Рисунок 4, 5 –Таяние два раза

Вывод: в ходе лабораторной работы мы научились использовать классы и методы в swift.