Лабораторна робота №6  
Розробка програм з використанням складних типів даних:   
двовимірних масивів та рядків

* 1. Мета:

Оволодіння навичками складання програм з використанням двовимірних масивів та рядків за допомогою засобів мови Kotlin.

* 1. Завдання:

1. Заповнити двовимірний масив згідно правила заданого рисунком. Розмір масиву вводити з клавіатури. Всі елементи, що потрапляють до зафарбованої області дорівнюють 1, всі інші - дорівнюють 0. Після заповнення, масив обов’язково вивести на екран. Рисунок для 2 варіанту зображено нижче.

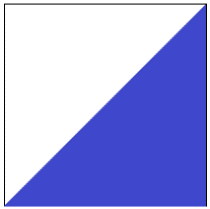


Рисунок до завдання

1. Згідно варіанту з завдання 6.1 знайти суму та середнє арифметичне значення елементів зафарбованої частини масиву. Для виконання завдання попередньо заповнити масив випадковими цілими числами з проміжку від -50 до 50 Результати обчислень вивести на екран.
2. Скласти програму для визначення кількості слів у рядку і виконати її в середовищі програмування. У завданні вважати, що рядок може містити лише літери, цифри та знаки пробілу. Слово - послідовність символів, що не містить пробілів.
   1. Практична частина
3. У програмі заповнено двовимірний масив згідно правила заданого рисунком. Розмір масиву вводиться з клавіатури. Всі елементи, що потрапляють до зафарбованої області дорівнюють 1, всі інші - дорівнюють 0. Після заповнення, масив виведений на екран.

fun var2z1() {

println("Введіть розмір масиву:")

val n = readln().toInt()

val matrix = Array(n) { IntArray(n) }

fillMatrix(matrix, n)

printMatrix(matrix)

}

fun fillMatrix(matrix: Array<IntArray>, n: Int) {

for (i in 0 until n) {

for (j in 0 until n) {

if (j >= n - i - 1) {

matrix[i][j] = 1

} else {

matrix[i][j] = 0

}

}

}

}

fun printMatrix(matrix: Array<IntArray>) {

println("Заповнений масив:")

for (row in matrix) {

for (element in row) {

print("$element ")

}

println()

}

}

1. У програмі згідно варіанту з завдання 6.1 шукається сума та середнє арифметичне значення елементів зафарбованої частини масиву. Для виконання завдання попередньо заповнено масив випадковими цілими числами з проміжку від -50 до 50 Результати обчислень виводяться на екран.

fun var2z2() {

println("Введіть розмір масиву:")

val n = readln().toInt()

val matrix = Array(n) { IntArray(n) }

fillMatrixRandomValues(matrix)

printMatrix(matrix)

val (sum, average) = calculateSumAndAverage(matrix, n)

println("Сума зафарбованої частини: $sum")

println("Середнє арифметичне зафарбованої частини: $average")

}

fun printMatrix(matrix: Array<IntArray>) {

println("Заповнений масив:")

for (row in matrix) {

for (element in row) {

print("$element ")

}

println()

}

}

fun fillMatrixRandomValues(matrix: Array<IntArray>) {

for (i in matrix.indices) {

for (j in matrix[i].indices) {

matrix[i][j] = Random.nextInt(-50, 51)

}

}

}

fun calculateSumAndAverage(matrix: Array<IntArray>, n: Int): Pair<Int, Double> {

var sum = 0

var count = 0

for (i in 0 until n) {

for (j in 0 until n) {

if (j >= n - i - 1) {

sum += matrix[i][j]

count++

}

}

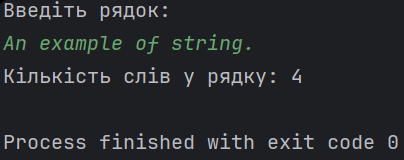
}

val average = if (count > 0) sum.toDouble() / count else 0.0

return Pair(sum, average)

}

1. Знизу розміщено код програми для визначення кількості слів у рядку і скриншот результатів з консолі. У завданні вважається, що рядок може містити лише літери, цифри та знаки пробілу. Слово - послідовність символів, що не містить пробілів.



Результати виконання програми у консолі

fun countWords(inputString: String): Int {

var count = 0

var inWord = false

for (char in inputString) {

if (char != ' ' && !inWord) {

count++

inWord = true

} else if (char == ' ') {

inWord = false

}

}

return count

}

* 1. Висновки

Оволодів навичками складання програм з використанням двовимірних масивів та рядків за допомогою засобів мови Kotlin .