object ArrayTools {

// Enctypted function takes input string and shift value from user create new encrypt string based on shift value.

fun encrypt(input: String, shift: Int): String {

val shifted = input.map { char ->

if (char.isLetter()) {

val base = if (char.isUpperCase()) 'A' else 'a'

val offset = (char - base + shift) % 26

(base + offset).toChar()

} else {

char

}

}

return shifted.joinToString("")

}

// arrayAvg function return average of array inputed by user

//fun <T : Number> arrayAvg(arr: Array<T>): Double {

// val sum = arr.sumOf { it.toDouble() }

//return sum / arr.size

// }

fun <T : Number> arrayAvg(arr: Array<T>): Double {

var sum = 0.0

for (element in arr) {

sum += element.toDouble()

}

return sum / arr.size

}

// arrayContains function check whether search value exist in array inputed by user

//fun <T> arrayContains(arr: Array<T>, searchValue: T): Boolean {

// return arr.contains(searchValue)

// }

fun <T> arrayContains(arr: Array<T>, searchValue: T): Boolean {

for (element in arr) {

if (element == searchValue) {

return true

}

}

return false

}

// reverseArray function reverse array inputed by user

fun <T> reverseArray(arr: Array<T>): Array<T> {

return arr.reversedArray()

}

}

fun main() {

// encrypt function input string and shift value

println("Enter string:")

val input = readLine() ?: ""

println("Enter string shift value:")

val shift = readLine()?.toIntOrNull() ?: 0

val encrypted = ArrayTools.encrypt(input, shift)

println("Encrypted string: $encrypted")

//arrayAvg and arrayContains function input array in double and search value in double

println("\n")

println("Enter array :")

val aryinput = readLine()

val values = aryinput?.split(" ")?.mapNotNull { it.toDoubleOrNull() }?.toTypedArray()

println("Enter array search value:")

val searchValue = readLine()?.toDoubleOrNull()

if (values != null && values.isNotEmpty()) {

val avg = ArrayTools.arrayAvg(values)

println("Average: $avg")

} else {

println("Invalid input!")

}

if (values != null && searchValue != null) {

val contains = ArrayTools.arrayContains(values, searchValue)

println("Array contains $searchValue: $contains")

} else {

println("Invalid input!")

}

// reverseArray function input take string array

println("\n")

println("Enter array:")

val revary = readLine()

val values1 = revary?.split(" ")?.toTypedArray()

if (values1 != null && values1.isNotEmpty()) {

val reversed = ArrayTools.reverseArray(values1)

println("Reversed array: ${reversed.contentToString()}")

} else {

println("Invalid input!")

}

}

Output :-

