import java.io.File  
import kotlin.math.absoluteValue  
  
fun main() {  
 val filePath = "C:\\Users\\Ghost Smoke\\Desktop\\LILIANI-5-0034\_10-19-2023\_12-50-00.2.txt" // Указываем путь к файлу  
 val targetWord = "акселерометра" // Замените на слово, которое нужно найти  
  
 // Читаем файл  
 val fileContent = File(filePath).*readLines*()  
  
 // Предполагаем, что сумма предыдущего значения равна 0  
 var previousSum = 0L  
 val targetWord2 = "Начало движения"  
  
 for (line in fileContent) {  
 // Ищем слово в строке  
 if (targetWord2 in line){  
 // Разбиваем строку на слова и находим числа  
 val numbers = line.*split*(" ")  
 .*filter* **{ it**.*toIntOrNull*() != null **}** // Оставляем только числа  
 .*map* **{ it**.*toInt*() **}** // Преобразуем строки в числа  
 *println*()  
 *println*(line)  
 }  
 if (targetWord in line) {  
  
 // Разбиваем строку на слова и находим числа  
 val numbers = line.*split*(" ")  
 .*filter* **{ it**.*toIntOrNull*() != null **}** // Оставляем только числа  
 .*map* **{ it**.*toInt*() **}** // Преобразуем строки в числа  
  
 // Суммируем найденные числа  
 val list = numbers.*toMutableList*()  
 list.removeFirst()  
  
 val currentSum : Int = list.*sum*()  
  
 val difference = (currentSum - previousSum).*absoluteValue* if (difference > 30) {  
 *println*()  
 *println*("line: $line")  
 *println*("Найдены числа: $list, их сумма: $currentSum")  
 *println*("Разница между суммами больше 30: $difference")  
 *println*("Значительные изменения происходят в: \n$line ")  
 }  
  
  
 // Обновляем предыдущее значение суммы  
  
 previousSum = currentSum.toLong()  
 }  
  
 }  
}

Описание к коду:

Используя дополнительные библиотеки java.io.File (работа с файлом) и kotlin.math.absoluteValue (для математических измерений).

val filePath = "C:\\Users\\Ghost Smoke\\Desktop\\LILIANI-5-0034\_10-19-2023\_12-50-00.2.txt" // Указываем путь к файлу с данными, может быть другим

val targetWord = “Искомое слово для ориентира”

val fileContent: = File(filePath).readLines() – чтение файла

var previousSum = 0L – т. к. сумма может быть большой – объявим переменную типа Long

val targetWord2 = “Начало движения”

for (line in fileContent) – цикл для поиска искомых строк

{

// Ищем слово в строке

if (targetWord2 in line){

// Разбиваем строку на слова и находим числа

val numbers = line.split(" ")

.filter { it.toIntOrNull() != null } // Оставляем только числа

.map { it.toInt() } // Преобразуем строки в числа

println()

println(line) – вывод строки с искомым словом

}

if (targetWord in line) – проверка на присутствие искомого слова в строке

{

// Разбиваем строку на слова и находим числа

val numbers = line.split(" ")

.filter { it.toIntOrNull() != null } // Оставляем только числа

.map { it.toInt() } // Преобразуем строки в числа

// Суммируем найденные числа

val list = numbers.toMutableList()

list.removeFirst() – удаление первого элемента (порядковый номер строки)

val currentSum : Int = list.sum() -

val difference = (currentSum - previousSum).absoluteValue – вычитание из нынешней суммы значений в строке предыдущее

if (difference > 30) – проверка на значительное изменение{

println()

println("line: $line")

println("Найдены числа: $list, их сумма: $currentSum")

println("Разница между суммами больше 30: $difference")

println("Значительные изменения происходят в: \n$line ")

}

// Обновляем предыдущее значение суммы

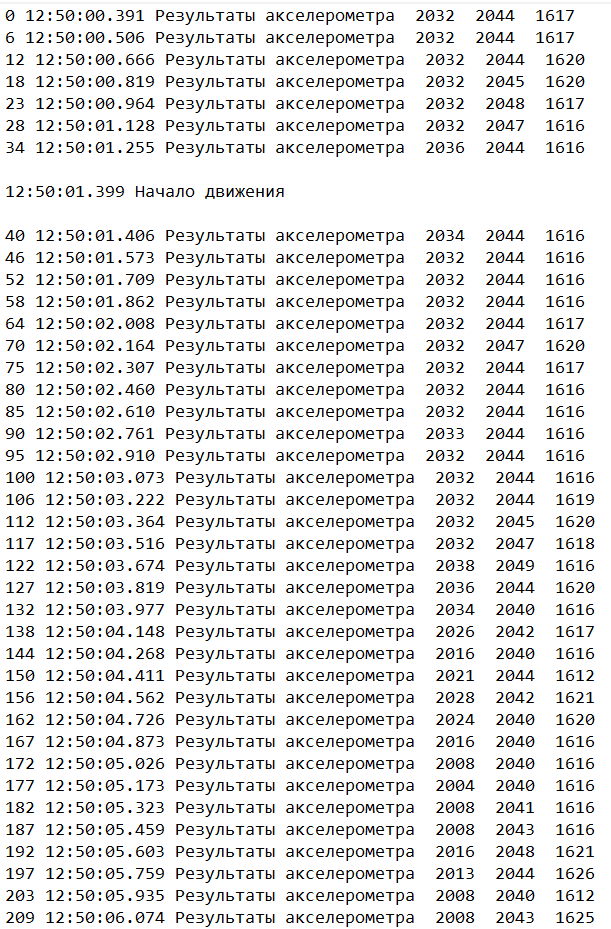
previousSum = currentSum.toLong() – перевод значение в тип Long

}

}

}

Данные из файла:



Получаемый результат:

