Одесский колледж компьютерных технологий «Сервер»

Отчет по практической работе №1

«Тема работы»

Выполнил:

студент группы К19.1

Фучило Богдан

Проверил:

Рудниченко Н.Д.

Одесса 2022

Содержание

[ВВЕДЕНИЕ 2](#_Toc461126358)

[ТЕОРИТИЧЕСКАЯ ЧАСТЬ 3](#_Toc461126359)

[ПРАКТИЧЕСКАЯ ЧАСТЬ 6](#_Toc461126360)

[ВЫВОД 14](#_Toc461126361)

[ЛИТЕРАТУРА 15](#_Toc461126362)

# ВВЕДЕНИЕ

Цель данной лабораторной работы просмотреть основные языковые особенности языка программирования java. Особенности построения логических выражений, типы данных, вывод данных, конкатенация строк, объектно-ориентированное программирование в java.

ТЕОРИТИЧЕСКАЯ ЧАСТЬ

Java — объектно-ориентированный язык программирования. В нем существуют классы (class) и объекты (object). Объекты в Java представляют собой экземпляры класса. В Java каждая строка кода, которая может выполняться, должна находиться внутри класса. Эта строка объявляет класс Main, модификатор доступа public означает что класс общедоступен и любой другой класс может получить доступ к нему. Несмотря на то, что язык Java объектно-ориентирован, не все типы — объекты.

Типы данных

Существуют так называемые примитивы (primitives). Вот список всех примитивов в Java:

byte (число, 1 байт)

short (число, 2 байта)

int (число, 4 байта)

long (число, 8 байтов)

float (число с плавающей точкой, 4 байта)

double (число с плавающей точкой, 8 байтов)

char (символ, 2 байта)

boolean (true (истина) или false (ложь), 1 байт)

Java — строго типизированный язык, это означает, что мы должны объявить переменные , прежде чем будем их использовать.

Чтобы объявить и присвоить число используйте следующий синтаксис:

int myNumber;

myNumber = 5;

= это оператор присваивания.

Вы можете объединить эти операции:

int myNumber = 5;

Чтобы объявить число с плавающей точкой, используйте следующий синтаксис:

double d = 4.5;

d = 3.0;

Если вы хотите использовать float, то:

float f = (float) 4.5;

Или:

float f = 4.5f

(f — более короткий способ объявить float)

Оператор break

Чтобы прервать последовательность повторений любого цикла в языке Java используется оператор break

**Пример.** Цикл прекратит повторяться, когда x превысит 99.

ß

for (float x = 1; ; x++ ) {

if (x > 99)

break;

sum += 1 / x;

}

Чтобы перейти к следующей итерации цикла, не завершая текущую, в языке С используется оператор continue.

Соглашение об именовании

Имена классов должны всегда начинаться с большой буквы (например, **ConnectionFactory**)

Имена пакетов должны состоять только из букв в нижнем регистре и цифр. При этом каждая составляющая имени должна начинаться с буквы (например, **org.apache.log4j**). Обычно имена пакетов начинаются с инвертированного имени домена компании-разработчика. Так из приведенного выше примера видно, что разработчиком проекта log4j является Apache Software Foundation ([www.apache.org](http://www.apache.org/)).

Названия методов и переменных должны начинаться с маленькой буквы и быть осмысленными. Каждое новое слово должно начинаться с большой буквы. Подчеркивания отсутствуют. (например, **insertToDatabase(),** **value**, **personName**).

Названия констант состоят из больших букв, цифр и знаков подчеркивания в качестве разделителей между словами (**MAX\_INTEGER**).

Управляющие конструкции

Отметим, что управляющие конструкции Java схожи с операторами языка С, но есть некоторые отличия.

***Оператор if / else***

Схема условного оператора такова:

**if (условие)**

**оператор 1**

**[else**

**оператор 2]**

Смысл условного оператора такой же как и в других языках программирования. В отличие от языка С условие всегда должно иметь логическое значение, поэтому пишут

if (i == 0) …

а не

if (!i) …

как в С

Булевы переменные

Переменные булевского типа (логические переменные) могут принимать одно из двух значений: «истина» или «ложь» и используются в языках программирования в операциях отношения (сравнения) и логических операциях. Так, результатом сравнения

**5 > 3**

будет «истина», а результатом сравнения

**8 < 1**

будет «ложь».

**boolean switch = true;**

ПРАКТИЧЕСКАЯ ЧАСТЬ

**Задание 1**

Напишите метод, который принимает на вход массив целых чисел (длиной 2 и более) и возвращает true, если в массиве каждый элемент больше или равен предыдущему. В противном случае он возвращает false.

Здесь представлен метод из класса ZadOne который отвечает за выполнение задания. По сути метод проверяет отсортирован ли массив, и в зависимости от результата возвращает true или false. Результат работы представлен на рисунке 1.

public static boolean arrayCheck(int[] array){  
 int i = 1;  
 for (;i < array.length-1 & array[i-1] <= array[i];i++);  
 if (i==array.length-1){  
 return true;  
 }  
 else return false;  
 }

}

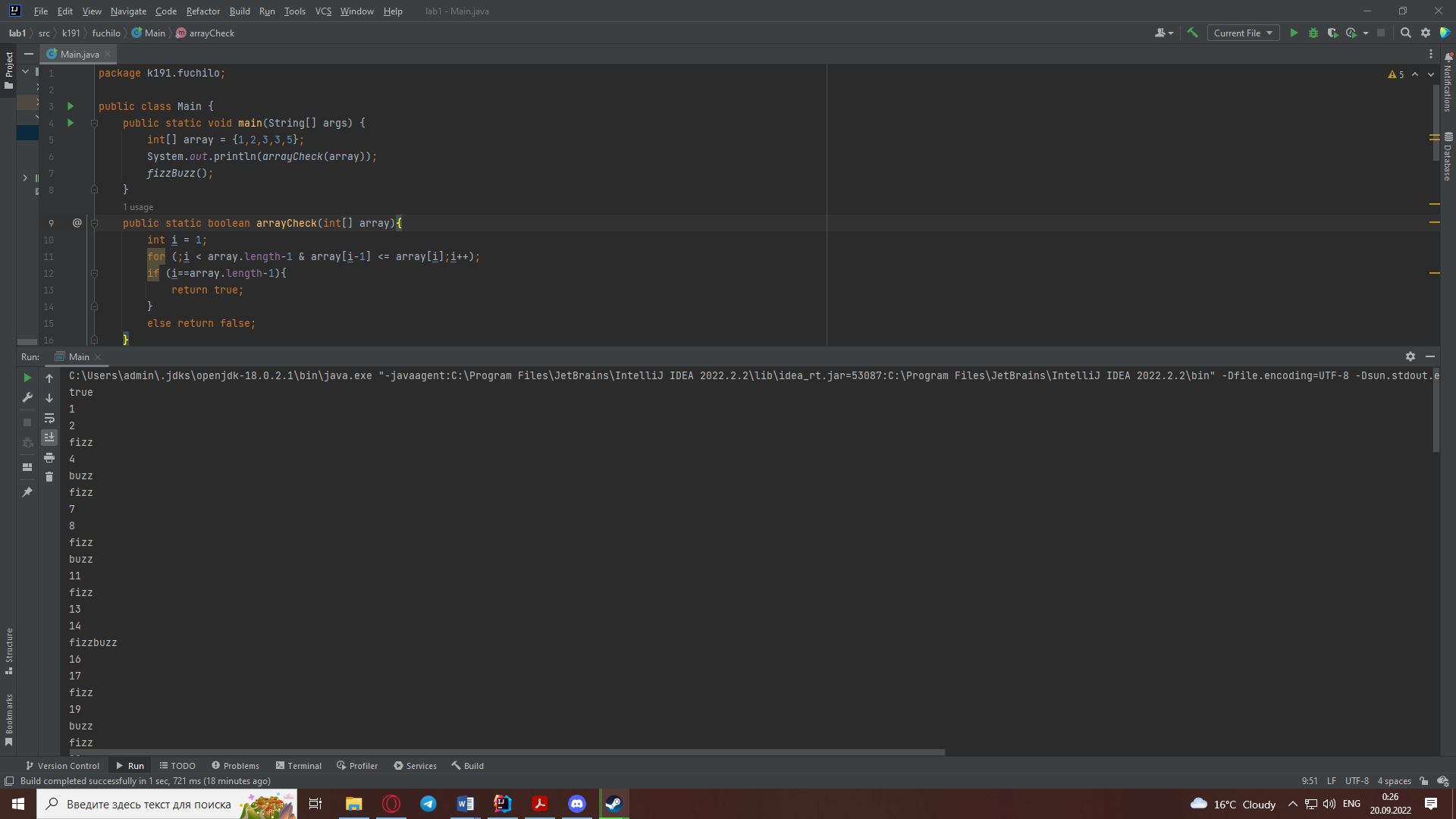


Рисунок 1.

**Задание 2**

Вы должны реализовать известную в Британии детскую игру FizzBuzz. Напишите метод, который выводит на экран числа от 1 до 100. При этом вместо чисел, кратных трем, программа должна выводить слово «Fizz», а вместо чисел, кратных пяти — слово «Buzz». Если число кратно и 3, и 5, то программа должна выводить слово «FizzBuzz».

Второй статический метод из класса ZadOne. Задание реализованно следующим образом(код ниже). Результат работы представлен на рисунке 2.

public static void fizzBuzz(){

for (int i = 1;i<=100;i++){

if (i % 3 == 0){

System.out.print("fizz");

}

if (i % 5 == 0){

System.out.print("buzz");

}

else if (i % 3 != 0 & i % 5 != 0)System.out.print(i);

System.out.print("\n");

}

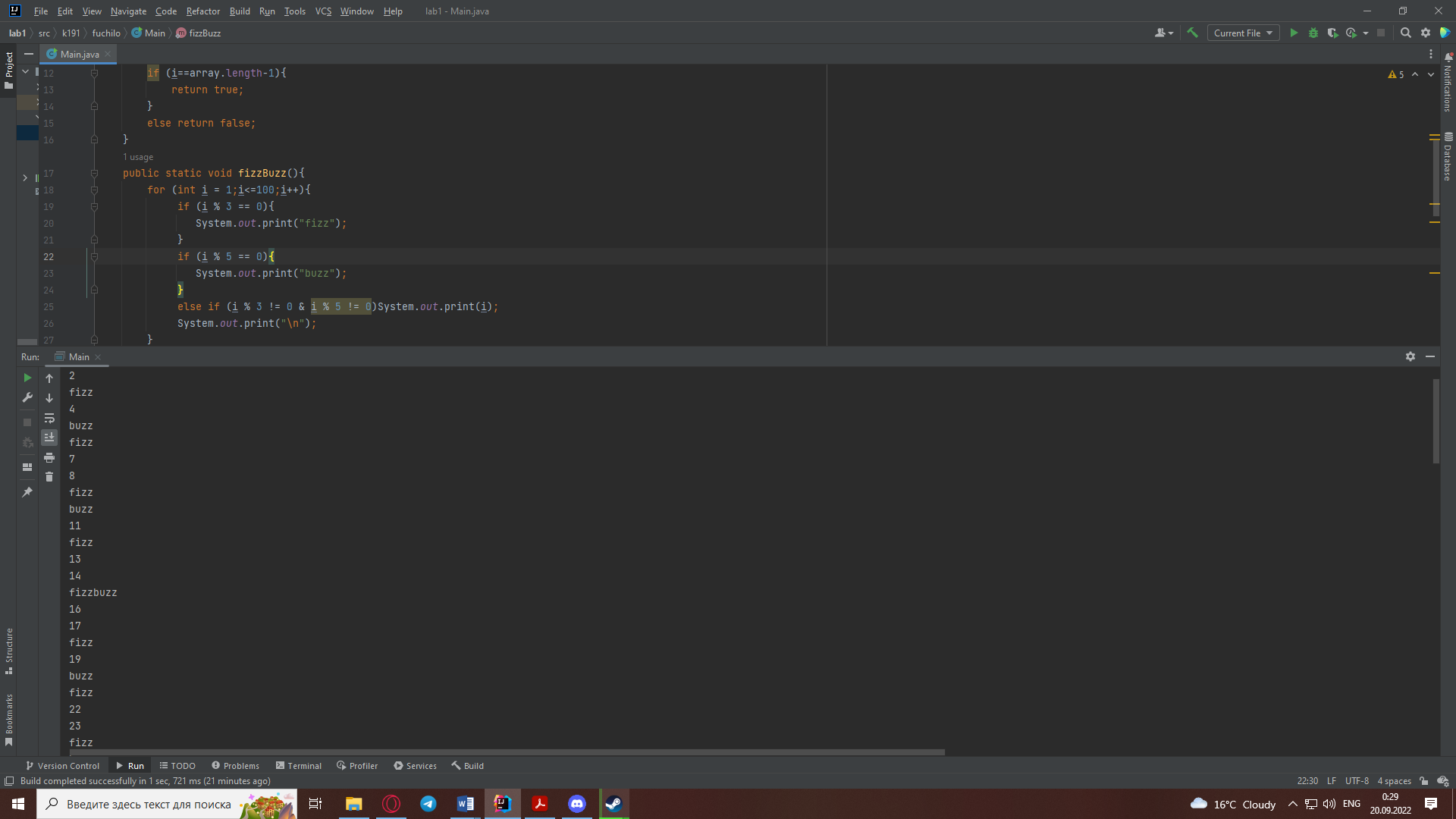


Рисунок 2.

**Задание 3**

Реализуйте метод, который принимает на вход массив целых чисел (длиной 2 или более) и возвращает true, если массив можно разделить так, чтобы сумма чисел в обеих частях была равной и false в противном случае. Например:

method({1, 1, 1, 2, 1}) → true // {1, 1 ,1} = {2, 1}

method ({2, 1, 1, 2, 1}) → false // не получается, т.к. {2, 1} < {1, 2, 1}, а {2, 1, 1} > {2, 1}

method ({10, 10}) → true // {10 = 10}

Реализация задания(код) находится ниже. Там же расположен результат работы программы на рисунке 3.

public static boolean arrayHalfSumChecker(int[] array) {

boolean f;

int length = array.length;

int firstHalf = 0, secondHalf = 0;

for(int i = 0; i<length; i++) {

if(i<length/2) {

firstHalf += array[i];

} else {

secondHalf += array[i];

}

}

if(firstHalf==secondHalf) {

f = true;

} else {

f = false;

}

return f;

}

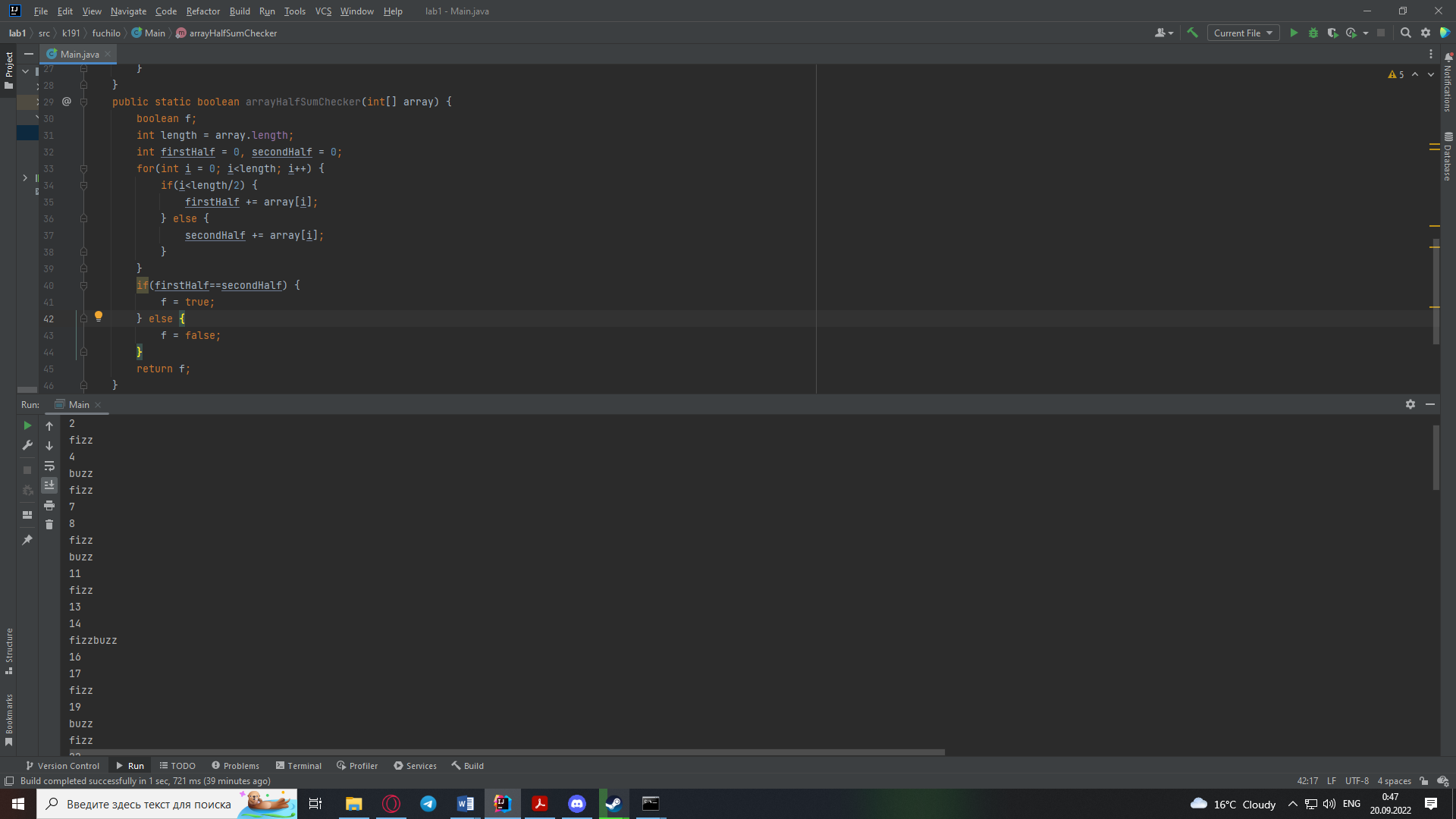


Рисунок 3.

ВЫВОДЫ

Выполнив первую лабораторную работу, были изучены синтаксические основы такого языка как Java. Были изучены операторы, вариации примитивных типов данных, логические операторы, циклы с пост и предусловием. Также было освоено, ПО необходимое для работы с языком Java, а именно IntelliJ IDEA.

ЛИТЕРАТУРА

1. <https://github.com/fuckkilla/Fuchilo_Java_Labs>