НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО”

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №1.2

з дисципліни

“ Основи Swift”

Виконав:

студент групи ІО-82

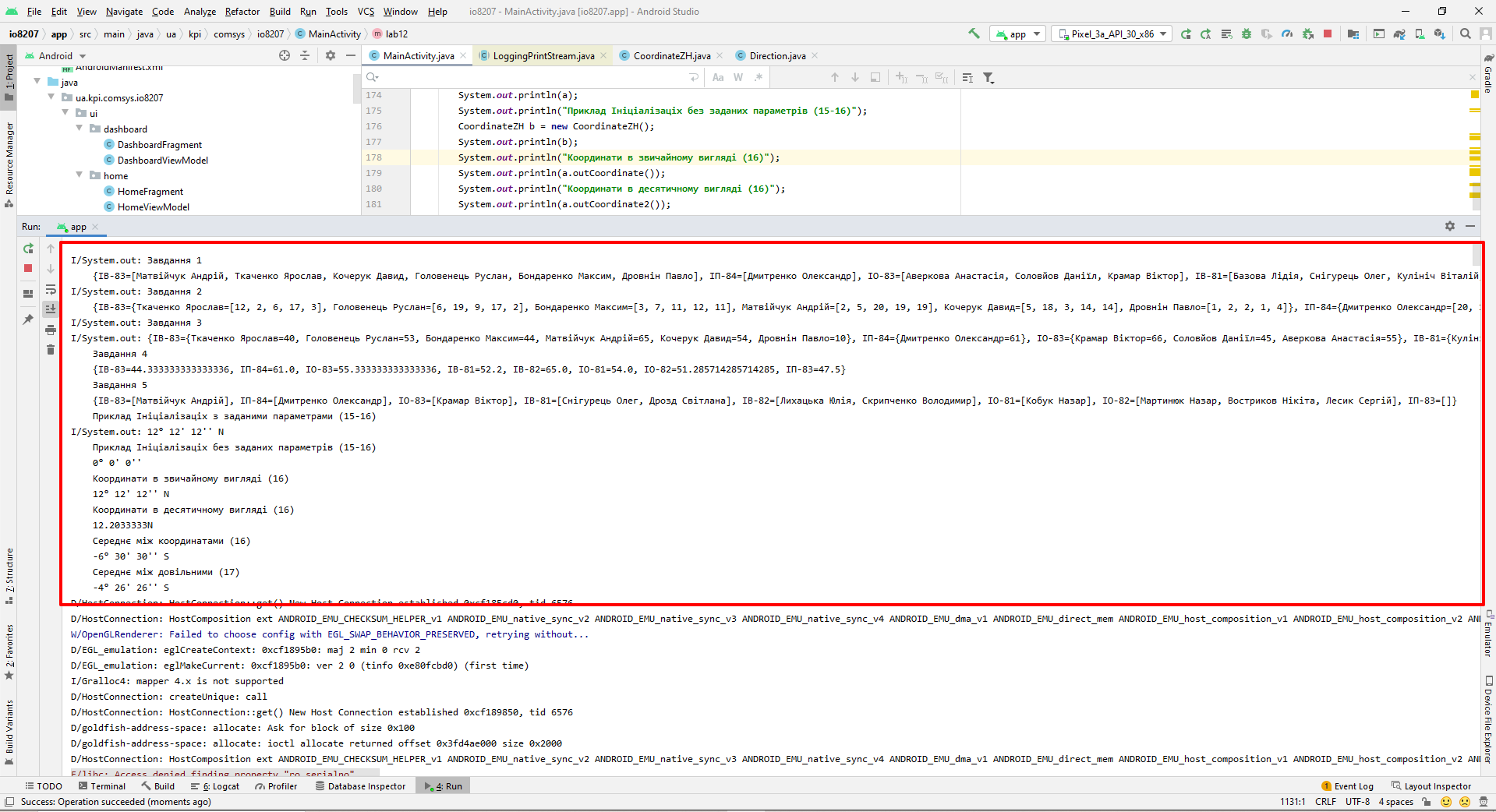
ЗК IO-8207

Галацин Захар

Київ 2021

**Варіант 2 (8207 % 2 + 1 = 2)**

**Скріншот роботи програми:**



**Результат роботи (тест):**

**Lab1.2 Start-------------------------------------------------**

**I/System.out: Завдання 1**

**I/System.out: {ІВ-83=[Матвійчук Андрій, Ткаченко Ярослав, Кочерук Давид, Головенець Руслан, Бондаренко Максим, Дровнін Павло], ІП-84=[Дмитренко Олександр], ІО-83=[Аверкова Анастасія, Соловйов Даніїл, Крамар Віктор], ІВ-81=[Базова Лідія, Снігурець Олег, Кулініч Віталій, Грабко Михайло, Дрозд Світлана], ІВ-82=[Лихацька Юлія, Скрипченко Володимир, Іванов Дмитро], ІО-81=[Рахуба Вероніка, Дудка Максим, Іванов Володимир, Кобук Назар], ІО-82=[Лесик Сергій, Ющенко Андрій, Мартинюк Назар, Роман Олександр, Востриков Нікіта, Тарасенко Юлія, Фещенко Кирил], ІП-83=[Мінченко Володимир, Жуков Михайло]}**

**I/System.out: Завдання 2**

**I/System.out: {ІВ-83={Ткаченко Ярослав=[17, 11, 16, 10, 8], Головенець Руслан=[2, 7, 8, 14, 2], Бондаренко Максим=[11, 11, 5, 19, 1], Матвійчук Андрій=[20, 20, 15, 9, 13], Кочерук Давид=[6, 10, 20, 1, 17], Дровнін Павло=[14, 9, 10, 9, 18]}, ІП-84={Дмитренко Олександр=[18, 10, 20, 4, 4]}, ІО-83={Крамар Віктор=[20, 13, 10, 3, 3], Соловйов Даніїл=[6, 14, 19, 20, 6], Аверкова Анастасія=[13, 16, 19, 6, 20]}, ІВ-81={Кулініч Віталій=[1, 8, 7, 7, 1], Базова Лідія=[10, 4, 14, 2, 2], Снігурець Олег=[19, 4, 8, 7, 4], Дрозд Світлана=[11, 14, 2, 14, 20], Грабко Михайло=[20, 8, 14, 20, 7]}, ІВ-82={Іванов Дмитро=[18, 7, 7, 1, 8], Лихацька Юлія=[12, 18, 1, 5, 2], Скрипченко Володимир=[8, 15, 10, 3, 16]}, ІО-81={Іванов Володимир=[16, 9, 14, 9, 4], Дудка Максим=[16, 8, 12, 13, 2], Кобук Назар=[5, 12, 6, 4, 15], Рахуба Вероніка=[18, 12, 7, 5, 10]}, ІО-82={Роман Олександр=[5, 11, 15, 12, 15], Мартинюк Назар=[2, 2, 9, 13, 4], Ющенко Андрій=[17, 9, 18, 19, 16], Тарасенко Юлія=[4, 10, 20, 14, 11], Востриков Нікіта=[1, 8, 7, 3, 7], Фещенко Кирил=[11, 13, 9, 11, 18], Лесик Сергій=[1, 18, 20, 15, 1]}, ІП-83={Мінченко Володимир=[8, 17, 17, 9, 18], Жуков Михайло=[3, 5, 20, 5, 1]}}**

**I/System.out: Завдання 3**

**{ІВ-83={Ткаченко Ярослав=62, Головенець Руслан=33, Бондаренко Максим=47, Матвійчук Андрій=77, Кочерук Давид=54, Дровнін Павло=60}, ІП-84={Дмитренко Олександр=56}, ІО-83={Крамар Віктор=49, Соловйов Даніїл=65, Аверкова Анастасія=74}, ІВ-81={Кулініч Віталій=24, Базова Лідія=32, Снігурець Олег=42, Дрозд Світлана=61, Грабко Михайло=69}, ІВ-82={Іванов Дмитро=41, Лихацька Юлія=38, Скрипченко Володимир=52}, ІО-81={Іванов Володимир=52, Дудка Максим=51, Кобук Назар=42, Рахуба Вероніка=52}, ІО-82={Роман Олександр=58, Мартинюк Назар=30, Ющенко Андрій=79, Тарасенко Юлія=59, Востриков Нікіта=26, Фещенко Кирил=62, Лесик Сергій=55}, ІП-83={Мінченко Володимир=69, Жуков Михайло=34}}**

**I/System.out: Завдання 4**

**{ІВ-83=55.5, ІП-84=56.0, ІО-83=62.666666666666664, ІВ-81=45.6, ІВ-82=43.666666666666664, ІО-81=49.25, ІО-82=52.714285714285715, ІП-83=51.5}**

**Завдання 5**

**I/System.out: {ІВ-83=[Ткаченко Ярослав, Матвійчук Андрій, Дровнін Павло], ІП-84=[], ІО-83=[Соловйов Даніїл, Аверкова Анастасія], ІВ-81=[Дрозд Світлана, Грабко Михайло], ІВ-82=[], ІО-81=[], ІО-82=[Ющенко Андрій, Фещенко Кирил], ІП-83=[Мінченко Володимир]}**

**Приклад Ініціалізаціх з заданими параметрами (15-16)**

**I/System.out: 12° 12' 12'' N**

**Приклад Ініціалізаціх без заданих параметрів (15-16)**

**0° 0' 0''**

**I/System.out: Координати в звичайному вигляді (16)**

**12° 12' 12'' N**

**Координати в десятичному вигляді (16)**

**12.2033333N**

**Середнє між координатами (16)**

**-6° 30' 30'' S**

**Середнє між довільними (17)**

**-4° 26' 26'' S**

**Лістинг коду**

**Повний код за посиланням:**

[**https://github.com/HalatsynZakhar/lab1.2\_mobile**](https://github.com/HalatsynZakhar/lab1.2_mobile)

**MainActivity.java**

**package** ua.kpi.comsys.io8207;  
  
**import** android.os.Bundle;  
  
**import** com.google.android.material.bottomnavigation.BottomNavigationView;  
  
**import** androidx.appcompat.app.AppCompatActivity;  
**import** androidx.navigation.NavController;  
**import** androidx.navigation.Navigation;  
**import** androidx.navigation.ui.AppBarConfiguration;  
**import** androidx.navigation.ui.NavigationUI;  
  
**import** java.util.ArrayList;  
**import** java.util.Arrays;  
**import** java.util.HashMap;  
**import** java.util.Random;  
**import** java.util.stream.IntStream;  
  
**public class** MainActivity **extends** AppCompatActivity {  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 BottomNavigationView navView = findViewById(R.id.***nav\_view***);  
 *// Passing each menu ID as a set of Ids because each  
 // menu should be considered as top level destinations.* AppBarConfiguration appBarConfiguration = **new** AppBarConfiguration.Builder(  
 R.id.***navigation\_home***, R.id.***navigation\_dashboard***, R.id.***navigation\_notifications***)  
 .build();  
 NavController navController = Navigation.*findNavController*(**this**, R.id.***nav\_host\_fragment***);  
 NavigationUI.*setupActionBarWithNavController*(**this**, navController, appBarConfiguration);  
 NavigationUI.*setupWithNavController*(navView, navController);  
  
 lab12();  
 }  
  
 **public void** lab12() {  
 System.***out***.println(**" \n\nLab1.2 Start-------------------------------------------------\n\n "**);  
  
  
 *//Завдання 1--------------------------------* String studentsStr = **"Дмитренко Олександр - ІП-84; Матвійчук Андрій - ІВ-83; Лесик Сергій - ІО-82;"** +  
 **" Ткаченко Ярослав - ІВ-83; Аверкова Анастасія - ІО-83; Соловйов Даніїл - ІО-83; "** +  
 **"Рахуба Вероніка - ІО-81; Кочерук Давид - ІВ-83; Лихацька Юлія - ІВ-82; "** +  
 **"Головенець Руслан - ІВ-83; Ющенко Андрій - ІО-82; Мінченко Володимир - ІП-83; "** +  
 **"Мартинюк Назар - ІО-82; Базова Лідія - ІВ-81; Снігурець Олег - ІВ-81; "** +  
 **"Роман Олександр - ІО-82; Дудка Максим - ІО-81; Кулініч Віталій - ІВ-81; "** +  
 **"Жуков Михайло - ІП-83; Грабко Михайло - ІВ-81; Іванов Володимир - ІО-81; "** +  
 **"Востриков Нікіта - ІО-82; Бондаренко Максим - ІВ-83; Скрипченко Володимир - ІВ-82; "** +  
 **"Кобук Назар - ІО-81; Дровнін Павло - ІВ-83; Тарасенко Юлія - ІО-82; "** +  
 **"Дрозд Світлана - ІВ-81; Фещенко Кирил - ІО-82; Крамар Віктор - ІО-83; Іванов Дмитро - ІВ-82"**;  
  
 HashMap<String, ArrayList> studentDict = **new** HashMap<>();  
  
 String[] studentsList = studentsStr.split(**"; "**);  
 String[] ListKey\_and\_Value;  
  
 **for** (**int** i = 0; i < studentsList.**length**; i++) {  
 ListKey\_and\_Value = studentsList[i].split(**" - "**);  
 ArrayList<String> listValue = **new** ArrayList<String>();  
 **if** (studentDict.containsKey(ListKey\_and\_Value[1])) {  
 listValue = studentDict.get(ListKey\_and\_Value[1]);  
 listValue.add(ListKey\_and\_Value[0]);  
 } **else** {  
 listValue.add(ListKey\_and\_Value[0]);  
 studentDict.put(ListKey\_and\_Value[1], listValue);  
 }  
 }  
 System.***out***.println(**"Завдання 1"**);  
 System.***out***.println(studentDict.toString());  
  
 *//Завдання 2--------------------------------* **int** min = 1;  
 **int** max = 20;  
 Random random = **new** Random();  
  
 HashMap<String, HashMap> studentDict2 = **new** HashMap<>();  
  
 String[] studentsList2 = studentsStr.split(**"; "**);  
 String[] ListKey\_and\_Value2;  
  
  
 **for** (**int** i = 0; i < studentsList2.**length**; i++) {  
 ListKey\_and\_Value2 = studentsList2[i].split(**" - "**);  
 HashMap<String, ArrayList> listValue2 = **new** HashMap<>();  
 ArrayList<Integer> listNumber = **new** ArrayList<Integer>();  
 listNumber.add(random.nextInt(max - min + 1) + min);  
 listNumber.add(random.nextInt(max - min + 1) + min);  
 listNumber.add(random.nextInt(max - min + 1) + min);  
 listNumber.add(random.nextInt(max - min + 1) + min);  
 listNumber.add(random.nextInt(max - min + 1) + min);  
 **if** (studentDict2.containsKey(ListKey\_and\_Value2[1])) {  
 HashMap<String, ArrayList> stud = **new** HashMap<>();  
 listValue2 = studentDict2.get(ListKey\_and\_Value2[1]);  
 listValue2.put(ListKey\_and\_Value2[0], listNumber);  
 } **else** {  
 HashMap<String, ArrayList> stud = **new** HashMap<>();  
 stud.put(ListKey\_and\_Value2[0], listNumber);  
 studentDict2.put(ListKey\_and\_Value2[1], stud);  
 }  
 }  
 System.***out***.println(**"Завдання 2"**);  
 System.***out***.println(studentDict2.toString());  
  
 *//Завдання 3--------------------------------* HashMap<String, HashMap> studentDict3 = **new** HashMap<>();  
  
 String[] studentsList3 = studentsStr.split(**"; "**);  
 String[] ListKey\_and\_Value3;  
  
  
 **for** (**int** i = 0; i < studentsList3.**length**; i++) {  
 ListKey\_and\_Value3 = studentsList3[i].split(**" - "**);  
 HashMap<String, Integer> listValue3 = **new** HashMap<>();  
 HashMap<String, ArrayList> HashNumber3 = **new** HashMap<>();  
 HashNumber3 = studentDict2.get(ListKey\_and\_Value3[1]);  
 ArrayList<Integer> sums = HashNumber3.get(ListKey\_and\_Value3[0]);  
 **int** sum = sums.get(0) + sums.get(1) + sums.get(2) + sums.get(3) + sums.get(4);  
 **if** (studentDict3.containsKey(ListKey\_and\_Value3[1])) {  
 HashMap<String, Integer> stud = **new** HashMap<>();  
 listValue3 = studentDict3.get(ListKey\_and\_Value3[1]);  
 listValue3.put(ListKey\_and\_Value3[0], sum);  
 } **else** {  
 HashMap<String, Integer> stud = **new** HashMap<>();  
 stud.put(ListKey\_and\_Value3[0], sum);  
 studentDict3.put(ListKey\_and\_Value3[1], stud);  
 }  
 }  
 System.***out***.println(**"Завдання 3"**);  
 System.***out***.println(studentDict3.toString());  
  
 *//Завдання 4--------------------------------* HashMap<String, Double> studentDict4 = **new** HashMap<>();  
  
 Double srznach = 0.0;  
 HashMap<String, Integer> temp = **new** HashMap<>();  
 **for** (String i : studentDict3.keySet()) {  
 temp = studentDict3.get(i);  
 **for** (Integer j : temp.values()) {  
 srznach += j;  
 }  
 srznach /= temp.size();  
 studentDict4.put(i, srznach);  
 srznach = 0.0;  
 }  
 System.***out***.println(**"Завдання 4"**);  
 System.***out***.println(studentDict4.toString());  
  
  
 *//Завдання 5--------------------------------* HashMap<String, ArrayList> studentDict5 = **new** HashMap<>();  
 HashMap<String, Integer> temp5 = **new** HashMap<>();  
 **for** (String i : studentDict3.keySet()) {  
 ArrayList<String> goodStudents = **new** ArrayList<String>();  
 temp5 = studentDict3.get(i);  
 **for** (String j : temp5.keySet()) {  
 **if** (temp5.get(j) >= 60) {  
 goodStudents.add(j);  
 }  
 }  
 studentDict5.put(i, goodStudents);  
 }  
 System.***out***.println(**"Завдання 5"**);  
 System.***out***.println(studentDict5.toString());  
  
 *//Завдання 6-12--------------------------------* System.***out***.println(**"Приклад Ініціалізаціх з заданими параметрами (15-16)"**);  
 CoordinateZH a = **new** CoordinateZH(12,12,12,**true**);  
 System.***out***.println(a);  
 System.***out***.println(**"Приклад Ініціалізаціх без заданих параметрів (15-16)"**);  
 CoordinateZH b = **new** CoordinateZH();  
 System.***out***.println(b);  
 System.***out***.println(**"Координати в звичайному вигляді (16)"**);  
 System.***out***.println(a.outCoordinate());  
 System.***out***.println(**"Координати в десятичному вигляді (16)"**);  
 System.***out***.println(a.outCoordinate2());  
 System.***out***.println(**"Середнє між координатами (16)"**);  
 System.***out***.println(a.outCoordinate3(-24, 48, 48, **true**));  
 System.***out***.println(**"Середнє між довільними (17)"**);  
 System.***out***.println(CoordinateZH.*outCoordinate4*(12, 12, 12, **true**, -20, 40, 40, **true**));  
 }  
}

**CoordinateZH.java**

**package** ua.kpi.comsys.io8207;  
  
**import** androidx.annotation.NonNull;  
  
**import** java.util.ArrayList;  
  
**import static** java.lang.Math.*round*;  
  
**public class** CoordinateZH **extends** Direction {  
  
 CoordinateZH(){  
 **this**.**gradus**=0;  
 **this**.**minuta**=0;  
 **this**.**secunda**=0;  
 }  
  
 @NonNull  
 @Override  
 **public** String toString() {  
 **return** outCoordinate();  
 }  
 CoordinateZH(**int** gradus, **int** minuta, **int** secunda, **boolean** dolgotqZeroShurotaone){  
 **if**(dolgotqZeroShurotaone){  
 **if** ((gradus >= -90) && (gradus <= 90)){  
 **this**.**gradus** = gradus;  
 }  
 }**else**{  
 **if** ((gradus >= -180) && (gradus <= 180)) {  
 **this**.**gradus** = gradus;  
 }  
 }  
 **if**((minuta >= 0) && (minuta <= 59)) {  
 **this**.**minuta** = minuta;  
 }  
 **if**((secunda >= 0) && (secunda <= 59)) {  
 **this**.**secunda** = secunda;  
 }  
  
 **this**.**dolgotqZeroShurotaone** = dolgotqZeroShurotaone;  
 }  
  
 String outCoordinate(){  
 String outText = **""**;  
 **if** (**this**.**dolgotqZeroShurotaone**){  
 **if**(**this**.**gradus** > 0){  
 outText = **"N"**;  
 }  
 **else if**(**this**.**gradus** <0){  
 outText = **"S"**;  
 }  
 } **else** {  
 **if**(**this**.**gradus** > 0) {  
 outText = **"E"**;  
 } **else if** (**this**.**gradus** < 0) {  
 outText = **"W"**;  
 }  
 }  
 **return this**.**gradus** + **"° "** + **this**.**minuta** + **"' "** + **this**.**secunda** + **"'' "** + outText;  
 }  
  
 String outCoordinate2() {  
 String outText = **""**;  
 **if** (**this**.**dolgotqZeroShurotaone**) {  
 **if** (**this**.**gradus** > 0) {  
 outText = **"N"**;  
 } **else if** (**this**.**gradus** < 0) {  
 outText = **"S"**;  
 }  
 } **else** {  
 **if** (**this**.**gradus** > 0) {  
 outText = **"E"**;  
 } **else if** (**this**.**gradus** < 0) {  
 outText = **"W"**;  
 }  
 }  
 **return this**.**gradus** + **"."** + *round*( (**this**.**minuta**\*60 + **this**.**secunda**)/3.6\*10000) + outText;  
 }  
  
 CoordinateZH outCoordinate3(**int** x, **int** y, **int** z, **boolean** shurota) {  
 **if** (shurota != **this**.**dolgotqZeroShurotaone**){  
 **return null**;  
 }  
  
 String outText = **""**;  
 **if** (**this**.**dolgotqZeroShurotaone**) {  
 **if** ((**this**.**gradus** + x) / 2 > 0) {  
 outText = **"N"**;  
 } **else if** ((**this**.**gradus** + x) / 2 < 0) {  
 outText = **"S"**;  
 }  
 } **else** {  
 **if** ((**this**.**gradus** + x) / 2 > 0) {  
 outText = **"E"**;  
 } **else if** ((**this**.**gradus** + x) / 2 < 0) {  
 outText = **"W"**;  
 }  
 }  
 CoordinateZH ret = **new** CoordinateZH((**this**.**gradus** + x) / 2, (**this**.**minuta** + y) / 2 , (**this**.**secunda** + z) / 2 , shurota);  
 **return** ret;  
 }  
  
  
 **public static** CoordinateZH outCoordinate4(**int** x1, **int** y1, **int** z1, **boolean** shurota1, **int** x2, **int** y2, **int** z2, **boolean** shurota2) {  
 **if** (shurota1 != shurota2){  
 **return null**;  
 }  
  
 String outText = **""**;  
 **if** (shurota1) {  
 **if** ((x1 + x2) / 2 > 0) {  
 outText = **"N"**;  
 } **else if** ((x1 + x2) / 2 < 0) {  
 outText = **"S"**;  
 }  
 } **else** {  
 **if** ((x1 + x2) / 2 > 0) {  
 outText = **"E"**;  
 } **else if** ((x1 + x2) / 2 < 0) {  
 outText = **"W"**;  
 }  
 }  
 CoordinateZH ret = **new** CoordinateZH((x1 + x2) / 2, (y1 + y2) / 2 , (z1 + z2) / 2 , shurota1);  
 **return** ret;  
 }  
  
}

**Direction.java**

**package** ua.kpi.comsys.io8207;  
  
**public class** Direction {  
 **int gradus**;  
 **int minuta**;  
 **int secunda**;  
 **boolean dolgotqZeroShurotaone**;  
}

Повний код за посиланням:

**Висновок**

У результаті виконання роботи я був ознайомлений з Android Studio. Було розроблено програму, що відображає інформацію на екрані, було створено меню з декількома елементами навігації. Засвоєно принципи роботи з лінійними структурами даних. Створено відповідну програму з виводом у консоль додатку