**Laboratorinis darbas Nr.1: Bazinės duomenų struktūros**

Atliko: Monika Kaltenytė, IFF-5/3, 2016 m. spalio 5 d.

# 1.1. Darbo užduotis

Pasinaudojant duotu projektu Lab1\_BazinesStrukturos, kuriame yra pateiktos bazinių struktūrų demonstracinės klasės su valdymo perjungimo metodais, reikia atlikti metodų tyrimus ir paaiškinti gautus rezultatus. Išbandyti sistemos NetBeans teikiamas galimybes.

# 1.2. Praktinio darbo tyrimas ir analizė

Pateikiami studento sukurtų metodų eksperimentai

|  |
| --- |
| ***Tiriamas metodas*** – B\_Labas double atstumasTarpTaškų(int x1, int y1, int x2, int y2) |
| double atstumasTarpTaškų(int x1, int y1, int x2, int y2){  double AB=0;  AB=(double) Math.sqrt(Math.pow((x2-x1), 2)+Math.pow((y2-y1), 2) );  return AB;  } |
| **Gauti rezultatai** |
| Atstumas tarp taškų 2.0 |
| **Komentarai:**  Panaudojamos matematinės funkcijos Math.sqrt ir Math.pow |
| ***Tiriamas metodas*** – B\_Labas.double koksVidurkis (int[] mas) |
| static double koksVidurkis(int[] mas){ int answer = 0;  for (int i = 0; i < mas.length; i++) { answer += mas[i];  }  return (double) answer / mas.length; } |
| **Gauti rezultatai** |
| 8.583 |
| **Komentarai:**  Praeinama pro visą masyvą ir surandamas jo elementų aritmetinis vidurkis |
| ***Tiriamas metodas*** – E\_MasyvuPvz int kiekŽaidėjųPelnėTaškus() |
| int kiekŽaidėjųPelnėTaškus(int [] mas){ int counter = 0;  for (int i = 0; i < mas.length; i++) { if(mas[i] == 0) counter++;  } return counter;  } |
| **Gauti rezultatai** |
| 3 |
| **Komentaras:**  Praeina pro įmestų taškų masyvą ir jei randa, kad neįmesta – prideda prie skaitiklio, o po to jį gražina. |

# 1.3. Pirmo kurso laboratorinio užduoties programa JAVA kalba

42. Žinomi prenumeratorių užsakyti dienraščiai. Žinomi duomenys apie dienraščius. Sudaryti užsakymų suvestinę.

Faile Gyventojai.txt yra miestelio prenumeratorių užsakymų sąrašas. Kiekvienoje eilutėje yra: pavardė ir vardas, dienraščio pavadinimas, užsakymo pradžios mėnuo ir keliems mėnesiams užsako. Jeigu gyventojas užsako kelis dienraščius arba kelis kartus per metus, tuomet jam yra skiriamos kelios eilutės ir nebūtinai iš eilės.

Faile Dienraščiai.txt yra duomenys apie leidinius. Kiekvienoje eilutėje yra: leidinio pavadinimas, vieno mėnesio kaina.

Rezultatų faile spausdinti suvestinį sąrašą už visus metus: leidinio pavadinimas, bendras užsakytų mėnesių skaičius, kaina ir pajamos. Pabaigoje spausdinti kiek gyventojai išleido pinigų spaudai per metus.

**Klasės:**

Lab1.java

Magazine.java

Resident.java

Subscription.java

**Metodai Subscription klasėje:**

public List<Resident> readResidents(String filePath)

public List<Magazine> readMagazines(String filePath)

public static double expenses(List<Magazine>magazines, List<Resident> residents)

public static void allMonths(List<Resident> residents, List<Magazine> magazines)

public static void allIncome(List<Resident> residents, List<Magazine> magazines)

public static void toConsole()

**Laukai :**

Magazine:

private String magazine;

private double price;

private int allMonths;

private double income;

Resident:

private String name;

private String title;

private int month;

private int duration;

**Metodai:**

public List<Resident> readResidents(String filePath) throws FileNotFoundException, IOException {

String line;

BufferedReader br = new BufferedReader(new FileReader(filePath));

while ((line = br.readLine()) != null) {

Resident Resident = new Resident(line);

residents.add(Resident);

}

return residents;

}

public List<Magazine> readMagazines(String filePath) throws FileNotFoundException, IOException {

String line;

BufferedReader br = new BufferedReader(new FileReader(filePath));

while ((line = br.readLine()) != null) {

Magazine magazine = new Magazine(line);

magazines.add(magazine);

}

return magazines;

}

public static double expenses(List<Magazine> magazines, List<Resident> residents) {

double allExpenses = 0;

for (Resident r : residents) {

for (Magazine m : magazines) {

if (r.getTitle().equals(m.getMagazine())) {

allExpenses += (r.getDuration() \* m.getPrice());

}

}

}

return allExpenses;

}

public static void allMonths(List<Resident> residents, List<Magazine> magazines) {

int temp = 0;

for (Magazine m : magazines) {

temp = 0;

for (Resident r : residents) {

if (m.getMagazine().equals(r.getTitle())) {

int d = r.getDuration();

temp += d;

}

m.setAllMonths(temp);

}

}

}

public static void allIncome(List<Resident> residents, List<Magazine> magazines) {

for (Magazine m : magazines) {

m.setIncome(m.getAllMonths() \* m.getPrice());

}

for (Magazine m : magazines) {

System.out.println(m.toString());

}

}

public static void toConsole() throws IOException {

Subscription ResidentsReader = new Subscription();

Subscription MagazinesReader = new Subscription();

List<Resident> residents = ResidentsReader.readResidents("F:\\2 KURSAS\\DUOMENŲ STRUKTŪROS\\Lab1(C-sharp)\\gyventojai.txt");

List<Magazine> magazines = MagazinesReader.readMagazines("F:\\2 KURSAS\\DUOMENŲ STRUKTŪROS\\Lab1(C-sharp)\\dienrasciai.txt");

Subscription.allMonths(residents, magazines);

Subscription.allIncome(residents, magazines);

System.out.println("Total expenses: " + Subscription.expenses(magazines, residents));

}

**Pradiniai Duomenys:**

Gyventojai.txt

Jonas Jonaitis;Kauno dienos;4;5

Petras Petraitis;Pasaka;7;3

Rasa Rasaitė;Naujienos;2;10

Lina Linaitė;Kauno dienos;3;7

Benas Benaitis;Pasaka;2;5

Toma Tomaitė;Pyragai;2;10

Milda Mildutė;Pasaka;12;1

Dienrasciai.txt

Kauno dienos;0.5

Pasaka;1.31

Naujienos;1.2

Pyragai;0.53

**Rezultatai:**

Title Price Months Income

Kauno dienos 0,50 12 6,00

Pasaka 1,31 9 11,79

Naujienos 1,20 10 12,00

Pyragai 0,53 10 5,30

Total expenses: 35.09

# Sisteminių klasių iš java.util bibliotekos išeities kodo analizė

Atlikau klasės String analizę. Ji turi tokius laukus:

private static final long serialVersionUID

private final char value[];

private int hash;

private static final ObjectStreamField[] serialPersistentFields

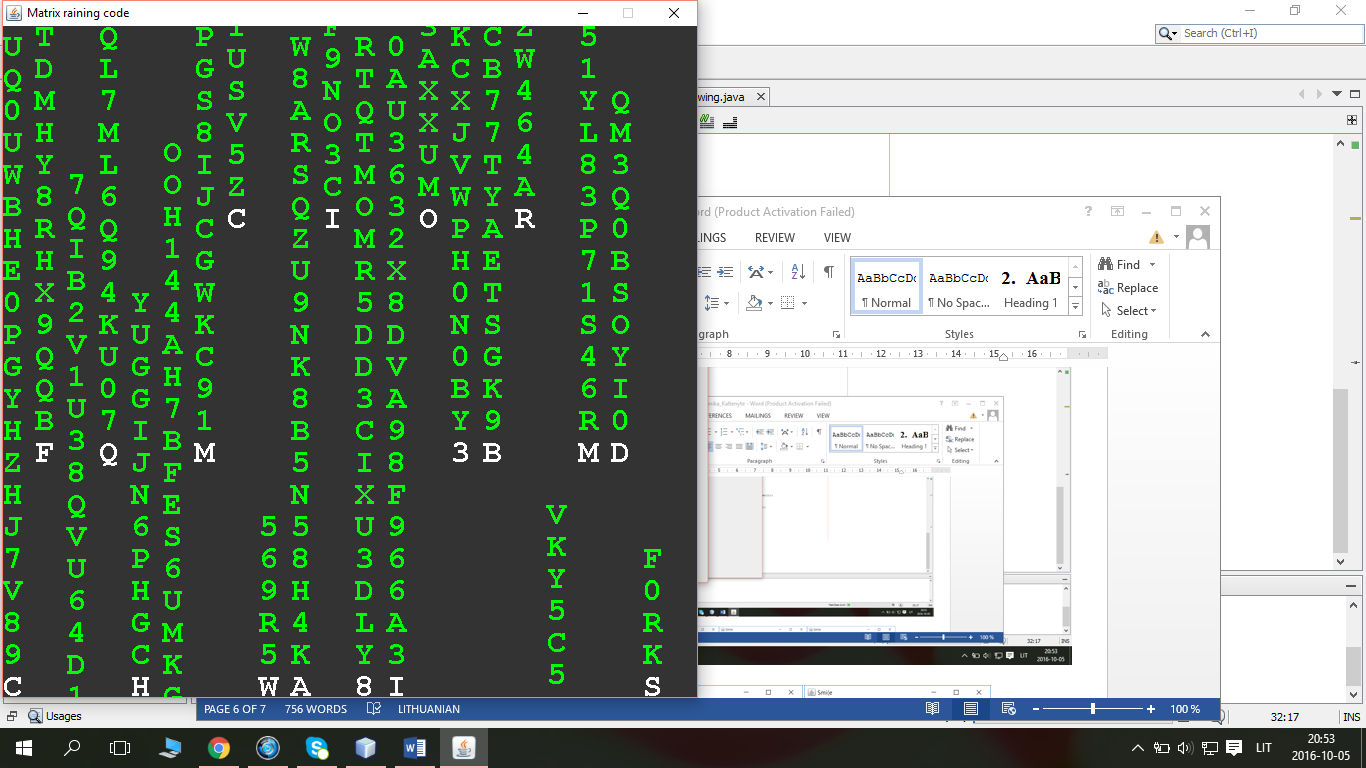
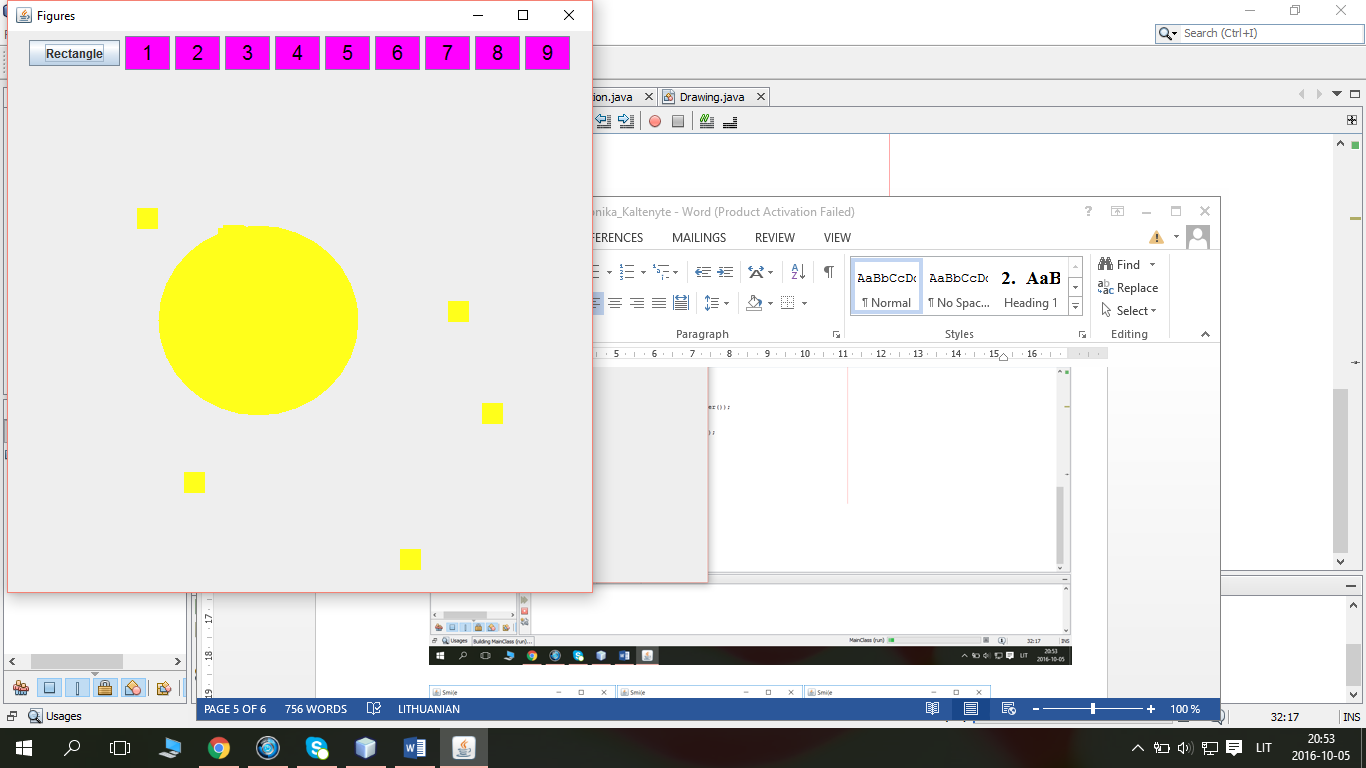
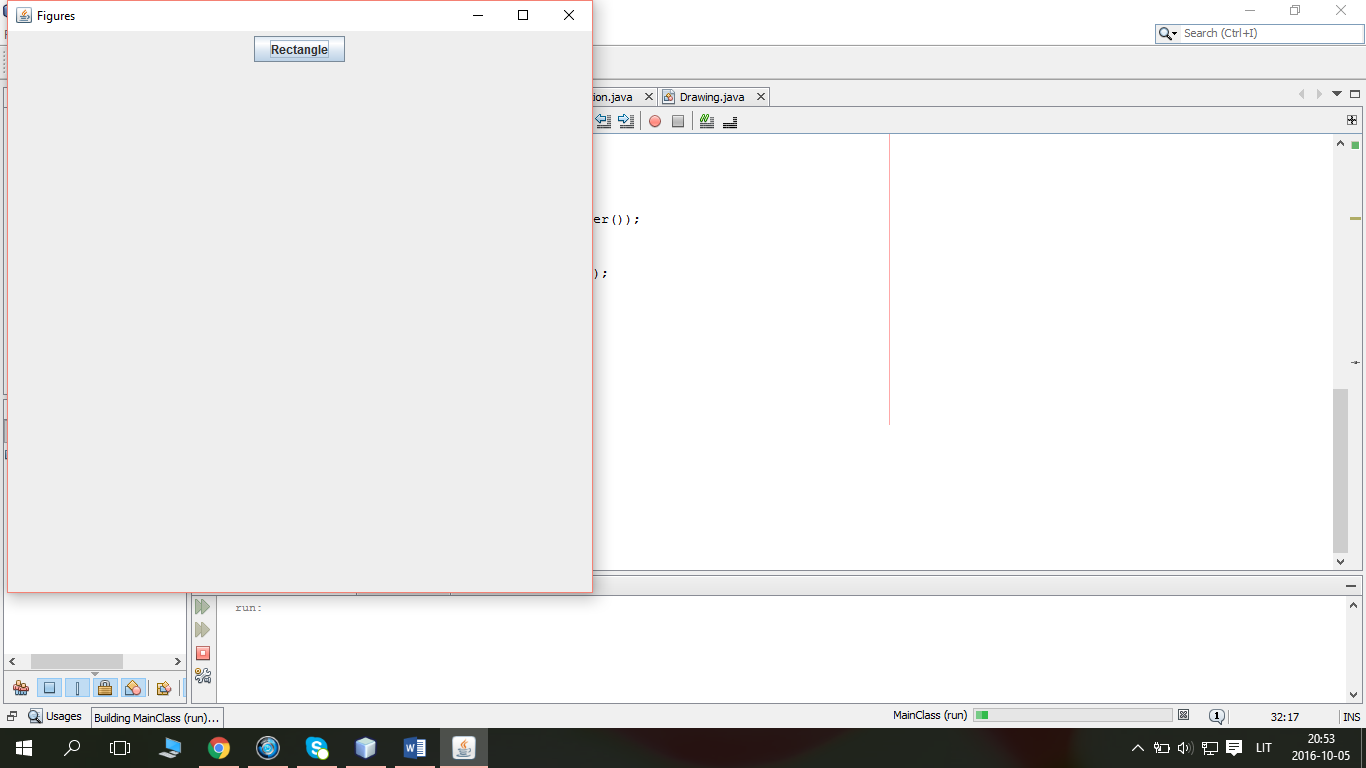
Neaiški yra kintamojo ObjectStreamField paskirtis ir serialVersionUID.

# Vizualios struktūros

Užduotis:

Sukurti vaizdinę vizualizaciją. Vartotojui ekrane spaudžiant pelę atsiranda apskritimas ir priklausomai nuo pelytės paspaudimų atsitiktinėje vietoje nupiešiamas stačiakampis. Pelytę paspaudus 9 kartus šis langas užveriamas ir atidaroma Matricos imitacija

Duomenys ir rezultatai:



# Išvados

Darant pirmąją laboratorinio darbo dalį didelių sunkumų pakeičiant C# laboratorinį į java neiškilo šiek tiek keblumų perrašant į java kalbą.

Didesnių sunkumų iškilo atliekant antrą dalį darant vizualias struktūras. Darant vizualias struktūras sunkiausia buvo padaryti, kad paspaudus mygtuką, įvyktų koks nors veiksmas. Teko pasigilinti kaip veikia vizualioms struktūroms būdingi metodai. Informacijos ieškota vizualizacijų pavyzdžiuose ir Google puslapiuose (pvz. stackoverflow.com)