KAUNO TECHNOLOGIJOS UNIVERSITETAS

Duomenų struktūros (P175B014) Projekto ataskaita

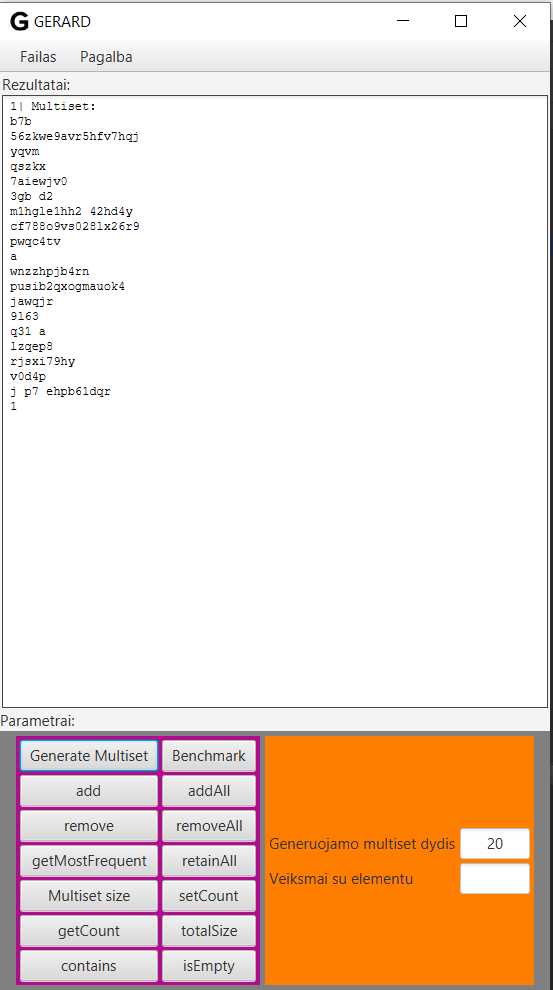
Atliko Arnas Švenčionis gr. IFF-8/11

Pasirinkta duomenų struktūra: Multiset.

Metodai ir jų veikimas:

Pradiniai duomenys:

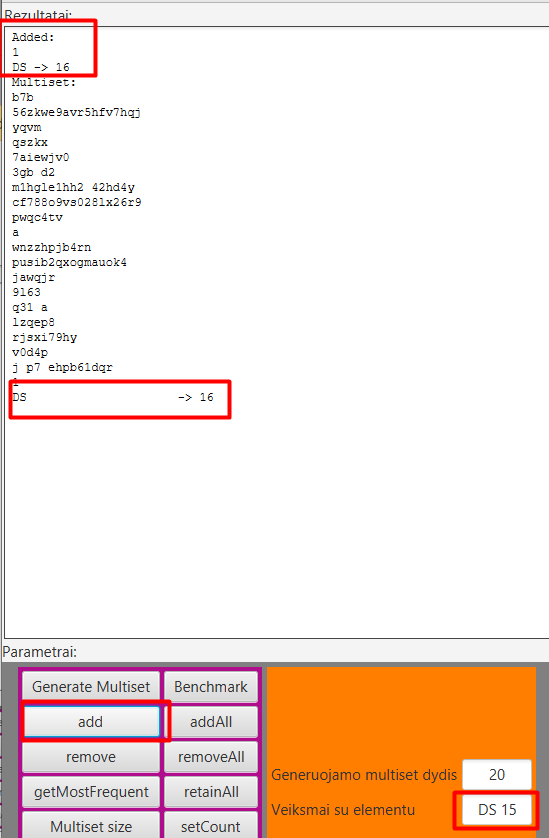
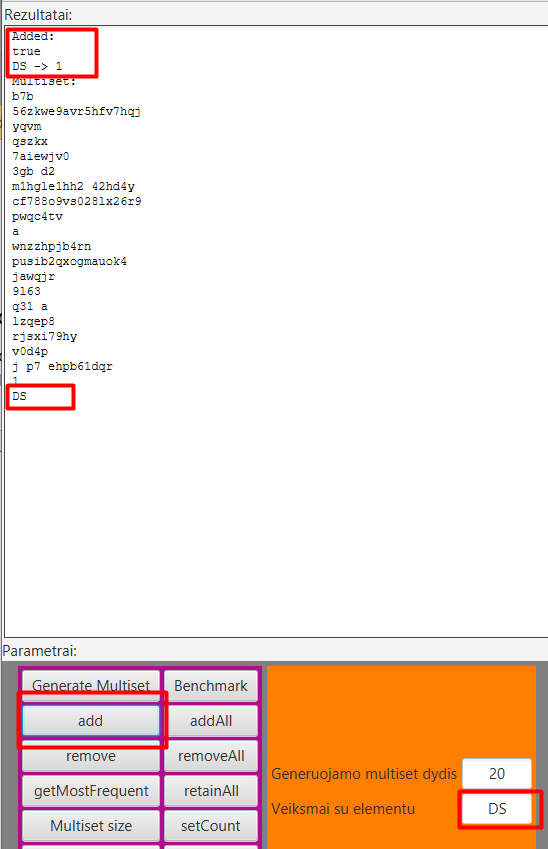
Sukurtu metodu yra sugeneruojamas atsitiktinių ilgių String Multiset



**Boolean add(E e) arba Boolean add(E e, int count)**

Į teksto laukelį parašius tik žodį, yra vykdomas add(E e) metodas, į sąrašą pridedantis vieną element pasirodymą. Parašius ir žodį, ir kiekį prie sąrašo yra pridedamas nurodytas kiekis.

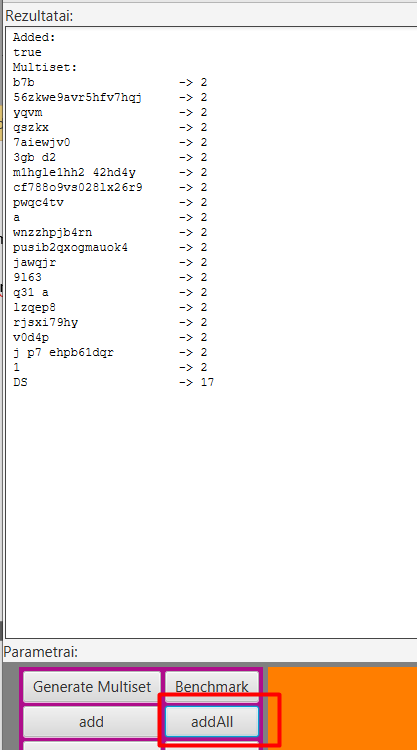
Rezultatai:



**Boolean addAll(Collection c)**

Į multiset yra pridedami visi kolekcijos elementai. Jei jie jau egzistuoja multiset, padidamas jų kiekis. Testavimo pavydyje yra pridedamas tas pats sąrašas prie šito.

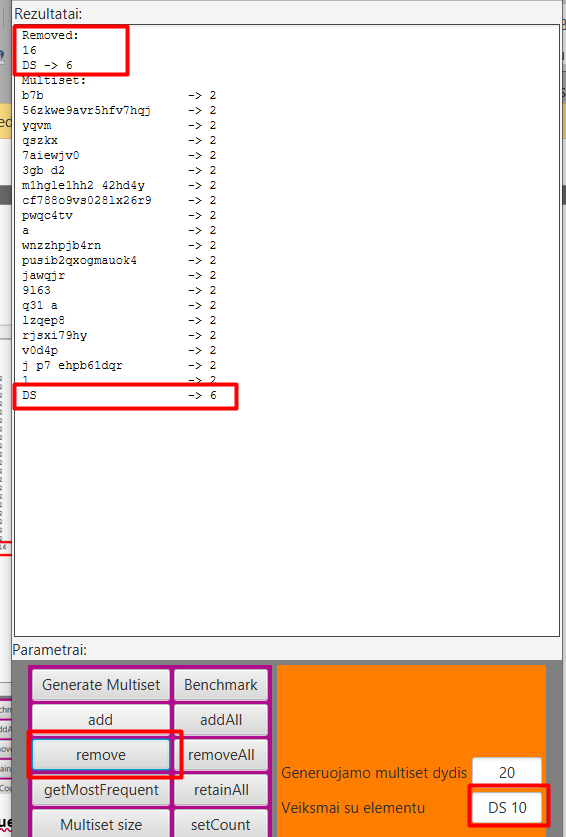
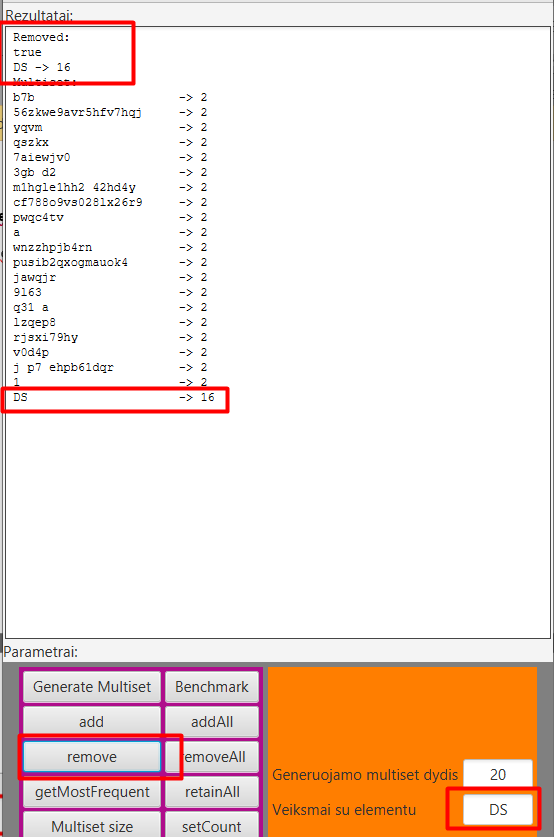
Rezultatai:



**Remove(E e)**

Ištrina vieną element e pasirodymą iš sąrašo jei teksto laukelyje yra nurodytas tik žodis. Jei yra nurodytas ir ždis ir skaičius, išrtinamas nurodytas kiekis pasikartojimų. Jei sąraše elementas pasikartoja tik vieną, arba mažiau nei nurodytą skairių kartų, jis yra visiškai ištrinamas iš sąrašo.

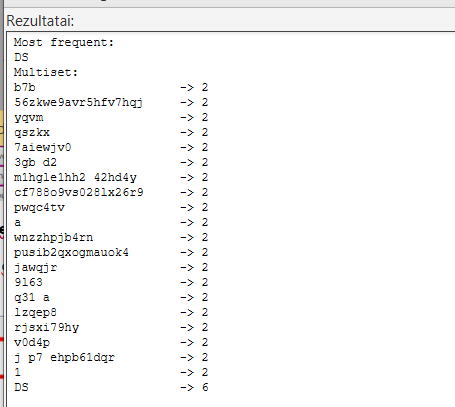
Rezultatai:



**E getMostFrequent();**

Randa ir grąžina elementą, su daugiausiai pasikartojimų

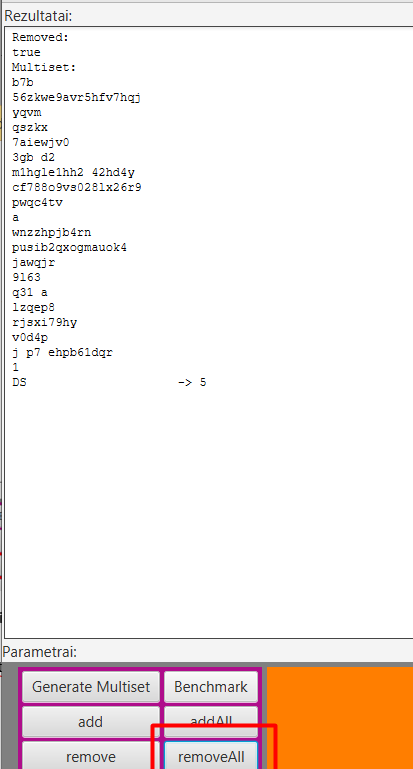
Rezultatai:



**removeAll(Collection c)**

Pašalina visus elementus, kurie yra nurodytoje kolekcijoje. Pavyzdyje metodui yra paduodama ta pati kolekcija.

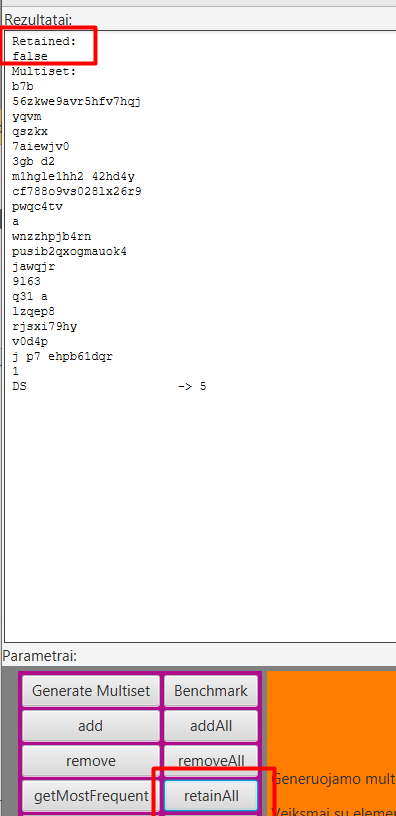
Rezultatai:



**retainAll(Collection c)**

Yra ištrinami visi elementai, kurie nėra nurodytoje kolekcijoje. Kadangi paduotas tas pats sąrašas, niekas nebuvo ištrinta, todėl metodas grąžina false.

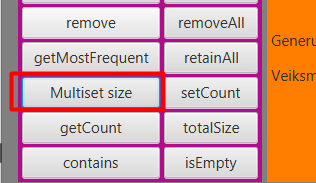
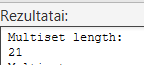
Rezultatai:



**Int multisetSize()**

Grąžina unikalių elementų skaičių multiset’e.

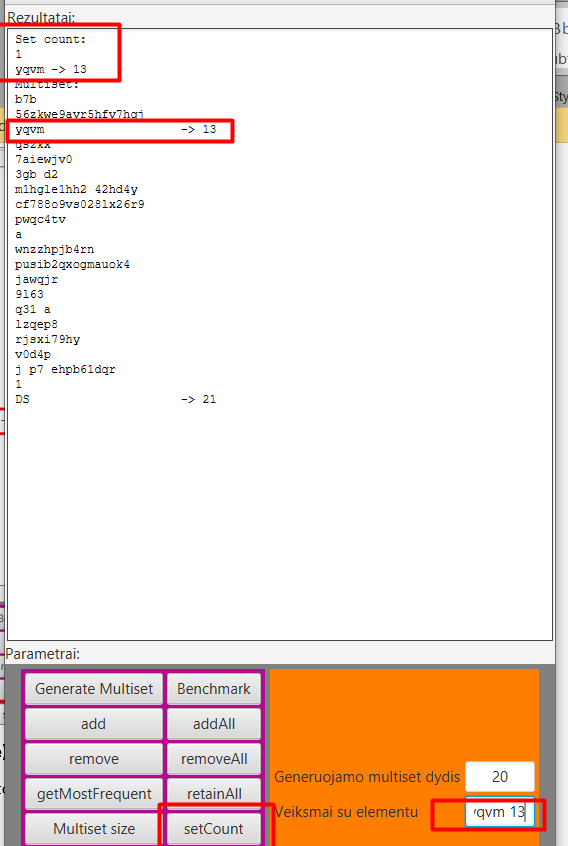
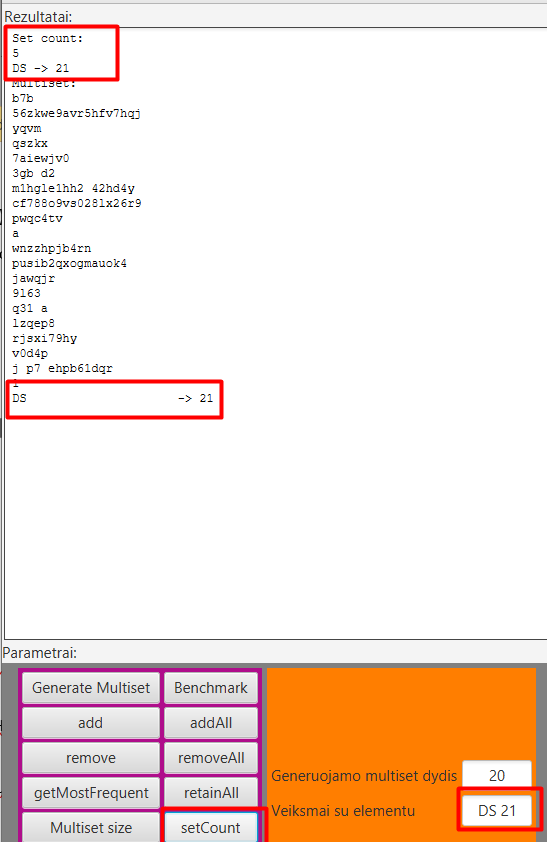
Rezultatai:



**setCount(E e, int count)**

Pakeičia nurodyto elemento count skaičių.

Rezultatai:



**Int getCount(E e)**

Grąžina elemento pasirodymų skaičių.

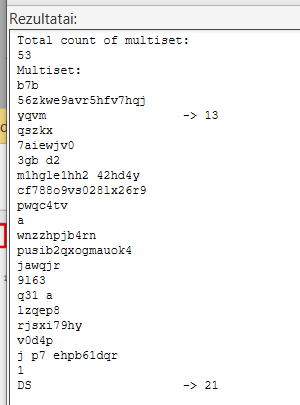
Rezultatai:



Int totalSize()

Grąžina visų elementų kiekių sumą.

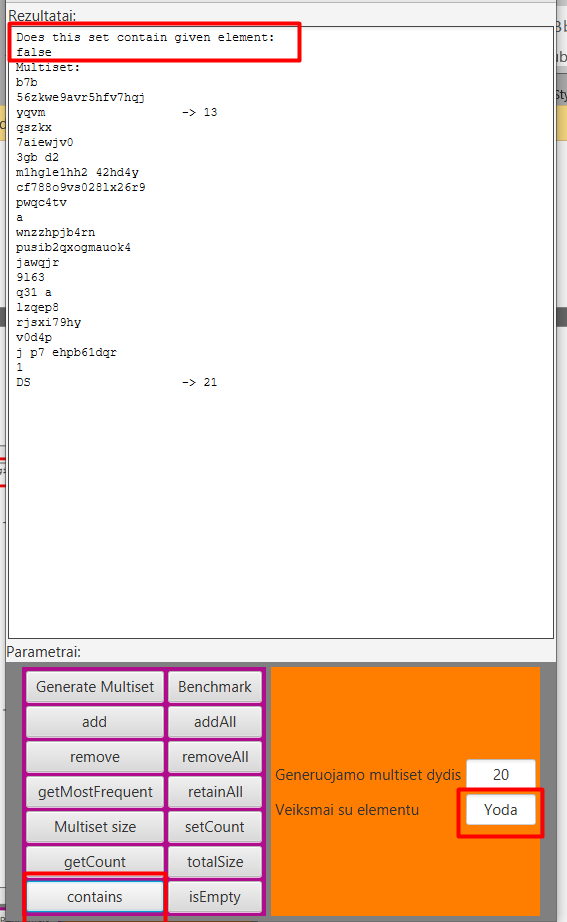
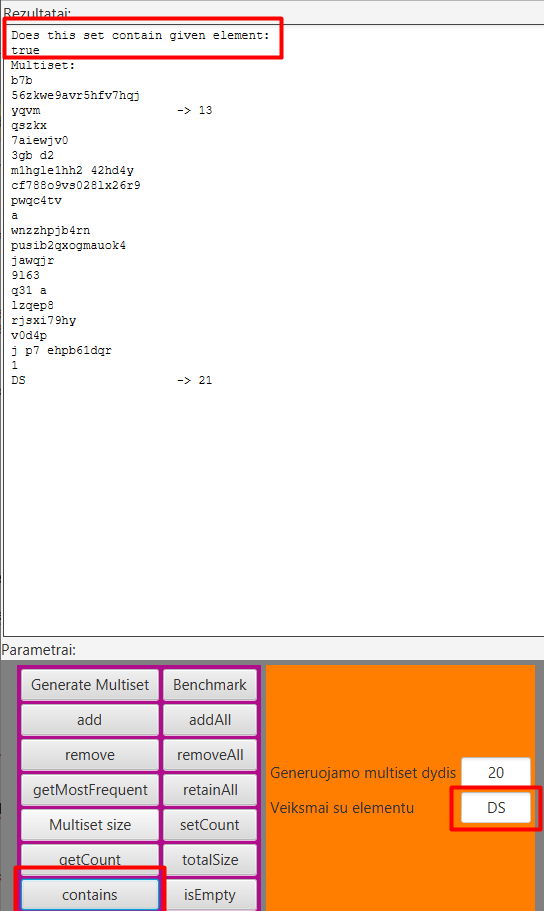
Rezultatas:



**boolean contains(E e)**

Grąžina true, jei elementas egzistuoja multiset’e

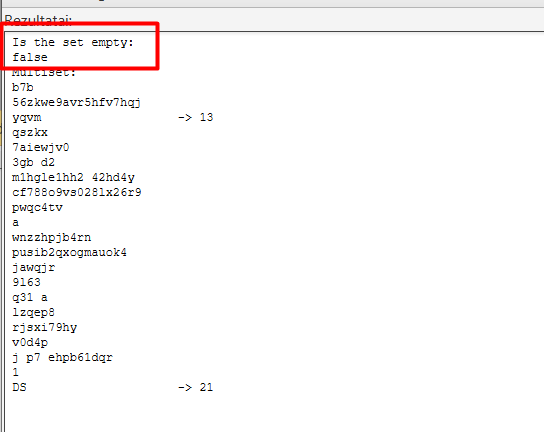
Rezultatas:



**Boolean isEmpty()**

Grąžina true, jei multiset yra tuščias.

Rezultatai:

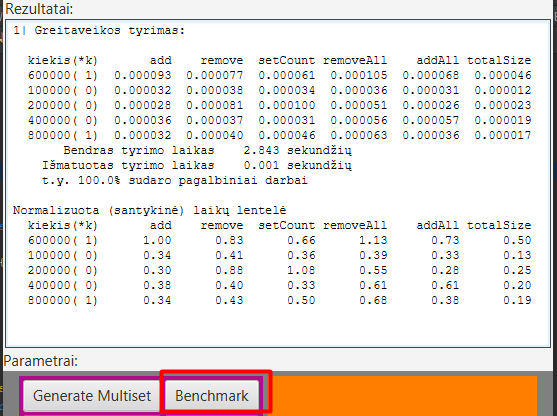


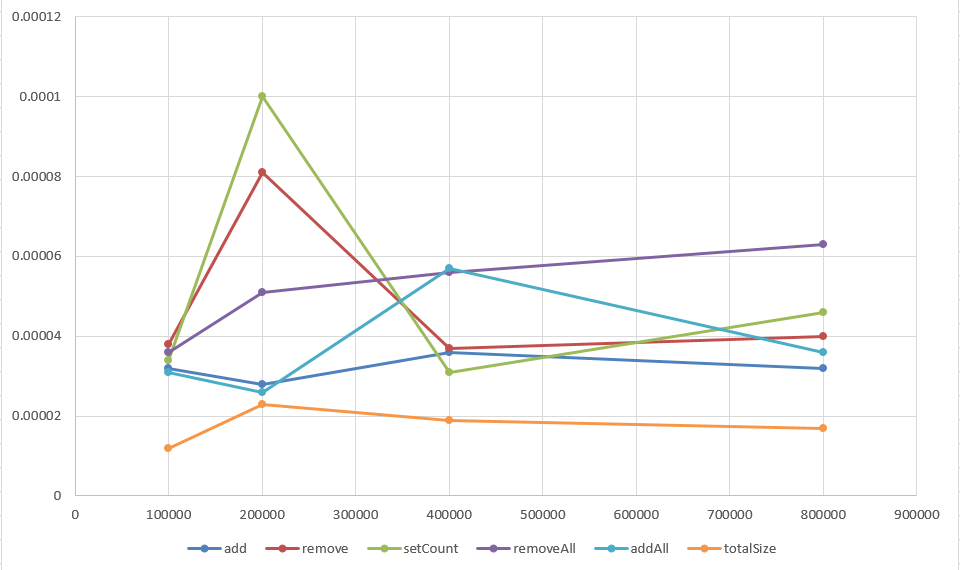
Greitaveika:

Palyginau pagrindinių metodų veikimo greičius.

Metodai, kurių greičius tyriau: add, remove, setCount, removeAll, addAll, totalSize.

Greitaveikos rezultatai:





Išvados:

Kadangi suradau gana nesudėtingą duomenų struktūrą, jai sukurti interface ir klasę labia sudėtinga nebuvo. Su interfeisais ir klasėmis susidūrėme visuose ankstesniuose laboratoriniuose darbuose, todėl iš naujo visko mokytis irgi nereikėjo. Sukūręs klasę sukūriau dar vieną klasę jos testavimui. Kadangi testavimai buvo sėkmingi, priėjau prie greitaveikos. Greitaveikos klasę pasiemiau iš praeitų laboratorinių darbų, tai ją liko tik pamodifikuoti. Sunkiausia dalis buvo gui sukūrimas, nors tam ištekliai irgi buvo duoti. Susidūriau su daug errorų, crashinimų ir pan. Tačiau galų gale pavyko pamodifikuoti praeitų laboratorinių darbų gui ir priskirti jį savai duomenų struktūrai.

Apytikslė darbo atlikimo trukmė: metodų ir interface kūrimas – 7 val., greitaveika – 3 val., GUI pakeitimai – 10 val.