**Laboratorinis darbas Nr.3: Aibes ir medžiai**

Atliko: Tadas Laurinaitis, IFF 6/8, 2017 12 18

# 1.1 Darbo užduotis

*Duota*: Projektas **Lab3\_AibesMedziai**, kuriame yra pateiktos toliau naudojamos sisteminės klasės, aibę aprašantys interfeisai **SortedSetADT**, **SortedSetADTx**, ir juos dvejetainiu paieškos medžiu ir AVL-medžiu realizuojančios demonstracinės klasės **BstSetKTU**, **AvlSetKTU**, **BstSetKTUx**, **AvlSetKTUx**.

*Atlikti*: Perkelkite reikiamas klases iš ankstesnių laboratorinių darbų individualioms užduotims spręsti ir atlikite reikalingas užduotis, susijusias su aibės realizacijų kūrimu bei panaudojimu.

*Tyrimo ir analizės dalis:*

1. Išnagrinėkite klasių **BstSetKTU** ir **AvlSetKTU** metodus, ištirkite jų panaudojimą operacijų su objektų aibe atlikimui.
2. Išnagrinėkite klasės **BstSetKTU** iteratoriaus klasės metodus, jų panaudojimą objektų peržiūrai.
3. Atlikite greitaveikos tyrimo eksperimentus su klase **Automobilis**.

*Individuali programavimo dalis:*

1. Pakete **lab3pavardė** sudarykite individualių elementų aibės panaudojimo klasę, kurioje būtų elementų aibės formavimas, elemento priklausomumo aibei patikrinimas, aibės elemento šalinimas ir pan. (3 metodai). Sukurtų metodų veikimą demonstruokite pateiktuose Swing arba JavaFX dialoguose arba sukurkite nuosavą, pasinaudodami paskaitų medžiaga.
2. Sudarykite individualių elementų panaudojimo klasės testus su skirtingais duomenų rinkiniais.
3. Parašykite metodą, skaičiuojantį vieną pasirinktą medžio charakteristiką: medžio aukštį, paieškos kelio ilgį vidutiniu ir blogiausiu atvejais, papildymo, šalinimo ir kitų operacijų vykdymo laiko priklausomybę nuo medžio aukščio.
4. Atlikite klasių **BstSetKTU** ir **AvlSetKTU** metodų greitaveikos tyrimą ir rezultatus palyginkite. Sudarykite vykdymo laikų grafikus ir atlikite rezultatų analizę.
5. Klasėje **BstSetKTU** realizuokite metodus *headSet(E e)*, *subSet(E e1, E e2)*, *tailSet(E e)* ir iteratoriaus metodą *remove()*. Galite pasiūlyti ir realizuoti kitų prasmingų darbui su rikiuota aibe metodų, pvz. metodai, realizuojantys dviejų aibių sąjungą, aibių sankirtą, patikrinantys ar dvi aibės yra lygios.
6. Realizuokite klasės **AvlSetKTU** metodus: *remove* ir *removeRecursive.*
7. Realizuoti individualiai nurodytus metodus **BstSetKTU** klasėje.
8. Atliekamas individualiai nurodytų metodų greitaveikos tyrimas.
9. Sunaudojamos atminties kiekio įvertinimas.

Pateikiami **studento sukurtų metodų** eksperimentai (viso 3-5 variantai)

|  |
| --- |
| ***Tiriamas metodas –* BstSetKTU.addall**() |
| public boolean addAll(BstSetKTU<?extends E> c){  Object[] objects = c.toArray();  for(Object obj:objects){  this.add((E)obj);  }  return true;  } |
| **Gauti rezultatai** |
| true |
| **Komentarai:**  Pridedami visi objektai is pasirinktos kolekcijos |
| ***Tiriamas metodas --* BstSetKTU.higher** () |
| public E higher(E e){  //List<E> ary = toEList();  Object[] objects = this.toArray();  E temp = null;  for(Object obj : objects){  if(e.compareTo((E) obj) < 0){  temp = (E) obj;  return temp;  }  }  return null;  } |
| **Gauti rezultatai** |
| **Komentarai:**  Grazinamas maziausias elementas aibeje didesnis uz nurodyta elementa |
| ***Tiriamas metodas --* BstSetKTU.** pollLast () |
| public E pollLast(){  E e = null;  Object[] objects = this.toArray();  if(this.size() <= 0){  e = null;  return e;  }  for(int i = 0; i < this.size();i++){  if(i == this.size()-1){  e = (E)objects[i];  this.remove((E)objects[i]);  return e;  }  }  return e;  } |
| **Gauti rezultatai** |
| **Komentarai:**  Grazinamas ir istrinamas paskutinis elementas |

# 1.3 Studento sukurtos klasės ir jų metodai

**Sprendimas:** aprašomos sudarytos klasės ir metodai *-> pilni kodai pateikiami kartu su projektu*.

**public class Kate** // Pagal šią klasę kuriamas Kates objektas

**Laukai:**

private String vardas;

private String veisle;

private String lytis;

private double amzius;

private double kaina; private int assists;

**Metodai:**

public String getName() //gražinamas Player vardas

public String getTeam() //gražinama Player komanda

public String getPosition() //gražinama Player pozicija

public String getChampion() //gražinamas Player čempionas

public int getKills() //gražinamas Player kill‘ų skaičius

public int getAssists() //gražinamas Player assist‘ų skaičius

**Metodai:**

public Kate create(String dataString)

public final void parse(String dataString)

public String validate()

public String toString()

public String getVardas()

public String getLytis()

public String getVeisle()

public double getAmzius()

public double getKaina()

public void setKaina(double kaina)

public int compareTo(Kate a)

**public class KaciuTestai**// Šioje klasėje atliekami veiksmai su katemis

**public class KaciuGamyba**// Šioje klasėje atliekami veiksmai su katemis

**public class KaciuApskaita**// Šioje klasėje atliekami parasytu metodu bandymai.

**Pradinis duomenų rinkinys:**

Sudaromas metodo create() pagalba

**Gauti rezultatai:**

# 1 Kaciu aibe:

# 2| Aibė papildoma: Kate 109=Prisma\_Sfinksas\_Mot:10.0 3600.0. Jos dydis: 1

# 3| Aibė papildoma: Kate 107=Mole\_Sfinksas\_Mot:5.0 2500.0. Jos dydis: 2

# 4| Aibė papildoma: Kate 108=Juode\_Abisinija\_Mot:6.0 7800.0. Jos dydis: 3

# 5| Aibė papildoma: Kate 105=Sniege\_Sfinksas\_Mot:3.0 3500.0. Jos dydis: 4

# 6| Aibė papildoma: Kate 101=Mikis\_Bengalas\_Vyr:4.0 4500.0. Jos dydis: 5

# 7| Aibė papildoma: Kate 106=Julia\_Abisinija\_Mot:4.0 500.0. Jos dydis: 6

# 8|

# 9| >──┬●Kate 109=Prisma\_Sfinksas\_Mot:10.0 3600.0

# │ ┌───●Kate 108=Juode\_Abisinija\_Mot:6.0 7800.0

# └──┼●Kate 107=Mole\_Sfinksas\_Mot:5.0 2500.0

# │ ┌───●Kate 106=Julia\_Abisinija\_Mot:4.0 500.0

# └──┼●Kate 105=Sniege\_Sfinksas\_Mot:3.0 3500.0

# └───●Kate 101=Mikis\_Bengalas\_Vyr:4.0 4500.0

# 10| Papildyta kaciu aibės kopija:

# 11| >──┬●Kate 109=Prisma\_Sfinksas\_Mot:10.0 3600.0

# │ ┌───●Kate 108=Juode\_Abisinija\_Mot:6.0 7800.0

# └──┼●Kate 107=Mole\_Sfinksas\_Mot:5.0 2500.0

# │ ┌───●Kate 106=Julia\_Abisinija\_Mot:4.0 500.0

# └──┼●Kate 105=Sniege\_Sfinksas\_Mot:3.0 3500.0

# │ ┌───●Kate 104=Angel\_Sfinksas\_Mot:2.0 12500.0

# │ ┌──┴●Kate 103=Siva\_Sfinksas\_Mot:6.0 5287.0

# │ ┌──┴●Kate 102=Jonas\_Abisinija\_Vyr:-1.0 3500.0

# └──┴●Kate 101=Mikis\_Bengalas\_Vyr:4.0 4500.0

# 12| Originalas:

# 13|

# >──┬●Kate 109=Prisma\_Sfinksas\_Mot:10.0 10000.0

# │ ┌───●Kate 108=Juode\_Abisinija\_Mot:6.0 7800.0

# └──┼●Kate 107=Mole\_Sfinksas\_Mot:5.0 2500.0

# │ ┌───●Kate 106=Julia\_Abisinija\_Mot:4.0 500.0

# └──┼●Kate 105=Sniege\_Sfinksas\_Mot:3.0 3500.0

# └───●Kate 101=Mikis\_Bengalas\_Vyr:4.0 4500.0

# 14| Ar elementai egzistuoja aibėje?

# 15| Kate 109=Prisma\_Sfinksas\_Mot:10.0 10000.0: true

# 16| Kate 107=Mole\_Sfinksas\_Mot:5.0 2500.0: true

# 17| Kate 108=Juode\_Abisinija\_Mot:6.0 7800.0: true

# 18| Kate 105=Sniege\_Sfinksas\_Mot:3.0 3500.0: true

# 19| Kate 101=Mikis\_Bengalas\_Vyr:4.0 4500.0: true

# 20| Kate 106=Julia\_Abisinija\_Mot:4.0 500.0: true

# 21| Kate 102=Jonas\_Abisinija\_Vyr:-1.0 3500.0: false

# 22| Kate 103=Siva\_Sfinksas\_Mot:6.0 5287.0: false

# 23| Kate 104=Angel\_Sfinksas\_Mot:2.0 12500.0: false

# 24|

# 25| Ar elementai egzistuoja aibės kopijoje?

# 26| Kate 109=Prisma\_Sfinksas\_Mot:10.0 10000.0: true

# 27| Kate 107=Mole\_Sfinksas\_Mot:5.0 2500.0: true

# 28| Kate 108=Juode\_Abisinija\_Mot:6.0 7800.0: true

# 29| Kate 105=Sniege\_Sfinksas\_Mot:3.0 3500.0: true

# 30| Kate 101=Mikis\_Bengalas\_Vyr:4.0 4500.0: true

# 31| Kate 106=Julia\_Abisinija\_Mot:4.0 500.0: true

# 32| Kate 102=Jonas\_Abisinija\_Vyr:-1.0 3500.0: true

# 33| Kate 103=Siva\_Sfinksas\_Mot:6.0 5287.0: true

# 34| Kate 104=Angel\_Sfinksas\_Mot:2.0 12500.0: true

# 35|

# 36| Elementų šalinimas iš kopijos. Aibės dydis prieš šalinimą: 9

# 37| Iš kaciu aibės kopijos pašalinama: Kate 102=Jonas\_Abisinija\_Vyr:-1.0 3500.0. Jos dydis: 8

# 38| Iš kaciu aibės kopijos pašalinama: Kate 101=Mikis\_Bengalas\_Vyr:4.0 4500.0. Jos dydis: 7

# 39| Iš kaciu aibės kopijos pašalinama: Kate 109=Prisma\_Sfinksas\_Mot:10.0 10000.0. Jos dydis: 6

# 40| Iš kaciu aibės kopijos pašalinama: Kate 108=Juode\_Abisinija\_Mot:6.0 7800.0. Jos dydis: 5

# 41| Iš kaciu aibės kopijos pašalinama: Kate 105=Sniege\_Sfinksas\_Mot:3.0 3500.0. Jos dydis: 4

# 42| Iš kaciu aibės kopijos pašalinama: Kate 103=Siva\_Sfinksas\_Mot:6.0 5287.0. Jos dydis: 3

# 43| Iš kaciu aibės kopijos pašalinama: Kate 104=Angel\_Sfinksas\_Mot:2.0 12500.0. Jos dydis: 2

# 44| Iš kaciu aibės kopijos pašalinama: Kate 102=Jonas\_Abisinija\_Vyr:-1.0 3500.0. Jos dydis: 2

# 45| Iš kaciu aibės kopijos pašalinama: Kate 107=Mole\_Sfinksas\_Mot:5.0 2500.0. Jos dydis: 1

# 46| Iš kaciu aibės kopijos pašalinama: Kate 106=Julia\_Abisinija\_Mot:4.0 500.0. Jos dydis: 0

# 47| Iš kaciu aibės kopijos pašalinama: Kate 107=Mole\_Sfinksas\_Mot:5.0 2500.0. Jos dydis: 0

# 48| Iš kaciu aibės kopijos pašalinama: Kate 109=Prisma\_Sfinksas\_Mot:10.0 10000.0. Jos dydis: 0

# 49|

# 50| Kaciu aibė su iteratoriumi:

# 51|

# 52| Kate 101=Mikis\_Bengalas\_Vyr:4.0 4500.0

# 53| Kate 105=Sniege\_Sfinksas\_Mot:3.0 3500.0

# 54| Kate 106=Julia\_Abisinija\_Mot:4.0 500.0

# 55| Kate 107=Mole\_Sfinksas\_Mot:5.0 2500.0

# 56| Kate 108=Juode\_Abisinija\_Mot:6.0 7800.0

# 57| Kate 109=Prisma\_Sfinksas\_Mot:10.0 10000.0

# 58|

# 59| Kaciu aibė AVL-medyje:

# 60|

# ┌───●Kate 109=Prisma\_Sfinksas\_Mot:10.0 10000.0

# ┌──┴●Kate 108=Juode\_Abisinija\_Mot:6.0 7800.0

# >──┼●Kate 107=Mole\_Sfinksas\_Mot:5.0 2500.0

# │ ┌───●Kate 106=Julia\_Abisinija\_Mot:4.0 500.0

# └──┼●Kate 105=Sniege\_Sfinksas\_Mot:3.0 3500.0

# └───●Kate 101=Mikis\_Bengalas\_Vyr:4.0 4500.0

# 61| Kaciu aibė su iteratoriumi:

# 62|

# 63| Kate 101=Mikis\_Bengalas\_Vyr:4.0 4500.0

# 64| Kate 105=Sniege\_Sfinksas\_Mot:3.0 3500.0

# 65| Kate 106=Julia\_Abisinija\_Mot:4.0 500.0

# 66| Kate 107=Mole\_Sfinksas\_Mot:5.0 2500.0

# 67| Kate 108=Juode\_Abisinija\_Mot:6.0 7800.0

# 68| Kate 109=Prisma\_Sfinksas\_Mot:10.0 10000.0

# 69|

# 70| Kaciu aibė su atvirkštiniu iteratoriumi:

# 71|

# 72| Kate 109=Prisma\_Sfinksas\_Mot:10.0 10000.0

# 73| Kate 108=Juode\_Abisinija\_Mot:6.0 7800.0

# 74| Kate 107=Mole\_Sfinksas\_Mot:5.0 2500.0

# 75| Kate 106=Julia\_Abisinija\_Mot:4.0 500.0

# 76| Kate 105=Sniege\_Sfinksas\_Mot:3.0 3500.0

# 77| Kate 101=Mikis\_Bengalas\_Vyr:4.0 4500.0

# 78|

# 79| Kaciu aibės toString() metodas:

# 80|

# Kate 101=Mikis\_Bengalas\_Vyr:4.0 4500.0

# Kate 105=Sniege\_Sfinksas\_Mot:3.0 3500.0

# Kate 106=Julia\_Abisinija\_Mot:4.0 500.0

# Kate 107=Mole\_Sfinksas\_Mot:5.0 2500.0

# Kate 108=Juode\_Abisinija\_Mot:6.0 7800.0

# Kate 109=Prisma\_Sfinksas\_Mot:10.0 10000.0

# 81|

# 82| Automobilių aibė DP-medyje:

# 83|

# ┌───●Kate 117=Zigfriedas\_Egzotas\_Vyr:3.0 2600.0

# ┌──┴●Kate 116=Zvalgas\_Egzotas\_Vyr:3.0 6500.0

# ┌──┴●Kate 115=Zydre\_Egzotas\_Mot:3.0 3200.0

# ┌──┴●Kate 114=Sniege\_Egzotas\_Mot:1.0 18000.0

# ┌──┴●Kate 113=Pacas\_Sfinksas\_Vyr:2.0 3500.0

# >──┴●Kate 112=Rimvydas\_Egzotas\_Vyr:3.0 7400.0

# 84| Išsiaiškinkite, kodėl medis augo tik į vieną pusę.

# 85|

# 86| Automobilių aibė AVL-medyje:

# 87|

# ┌───●Kate 123=Zigfriedas\_Egzotas\_Vyr:3.0 2600.0

# ┌──┴●Kate 122=Zvalgas\_Egzotas\_Vyr:3.0 6500.0

# >──┼●Kate 121=Zydre\_Egzotas\_Mot:3.0 3200.0

# │ ┌───●Kate 120=Sniege\_Egzotas\_Mot:1.0 18000.0

# └──┼●Kate 119=Pacas\_Sfinksas\_Vyr:2.0 3500.0

# └───●Kate 118=Rimvydas\_Egzotas\_Vyr:3.0 7400.0

# 88| Pasikartojančios kaciu veisles:

# Abisinija

# Sfinksas

**1.4 Greitaveikos Tyrimas ir rezultatai:**

Laboratorinio darbo užduotyje buvo duoti du tyrimai, kuriuos reikėjo atlikti: 1) Palyginti klasių TreeSet<Integer> ir HashSet<Integer> metodus remove(Object o); 2) ) Palyginti klasių TreeSet<Integer> ir HashSet<Integer> metodus containsAll(Collection<?> c);

**Greitaveikos tyrimo rezultatai:**

Greitaveikos tyrimas:

1. kiekis(\*k) remove remove containsAll containsAll
2. 10000( 1) 0.0081 0.0022 0.0025 0.0004
3. 20000( 2) 0.0027 0.0025 0.0025 0.0015
4. 40000( 4) 0.0033 0.0016 0.0048 0.0008
5. 80000( 8) 0.0066 0.0038 0.0071 0.0018
6. Bendras tyrimo laikas 1.873 sekundžių
7. Išmatuotas tyrimo laikas 0.052 sekundžių
8. t.y. 97.2% sudaro pagalbiniai darbai
9. Normalizuota (santykinė) laikų lentelė
10. kiekis(\*k) remove remove containsAll containsAll
11. 10000( 1) 1.00 0.27 0.31 0.05
12. 20000( 2) 0.33 0.31 0.30 0.19
13. 40000( 4) 0.41 0.20 0.59 0.10
14. 80000( 8) 0.81 0.46 0.87 0.22

# Išvados

Labai patogu, medžių ir aibių pagalba talpinti ir atvaizduoti įvarius duomenis.

Bendros laboratorinio darbo atlikimo laiko sąnaudos buvo 7 valandų.