

L2 užduotis

L2 privalomoji dalis

1. Išnagrinėkite Lab2_AibesMedziai projekte pateiktą programinį kodą.
2. (5 balai) Realizuokite nerealizuotus programinio kodo metodus ir juos ištetsuokite (testus galite inicijuoti tiek per projekto vartotojo sąsają, tiek per konsolę).

Klasėje BstSet reikia realizuoti šiuos metodus:

- remove(E element);
- removeRecursive(E element, BstNode<E> node) - pagalbinis remove metodas.
- Iteratoriaus metodą remove();
- addAll(Set<E> set) - patalpinti visus set elementus į aibę, jeigu abi aibės turi tą patį elementą jis nėra dedamas.
- containsAll(Set<E> set) - patikrinti ar aibėje egzistuoja visi elementai, esantys aibėje set.
- retainAll(Set<E> set) - išrinti elementus, kurių nėra aibėje set.
- headSet(E e) - grąžinamas aibės poaibis iki elemento e.
- tailSet(E e) - grąžinamas aibės poaibis nuo elemento e.
- subSet(E element1, E element2) - grąžinamas aibės poaibis nuo elemento element1 iki element2.

Klasėje AvlSet reikia realizuoti šiuos metodus:

- remove();
- removeRecursive(E element, AVLNode<E> node) - pagalbinis remove metodas.

3. (3 balai) Atlikite greitaveikos testavimą (naudokite JMH) su viena metodų pora iš žemiau pateiktos lentelės:

| Variantas | Metodas 1 | Metodas 2 |
|-----------|-----------------------------|--|
| 1 | Class BstSet: remove() | Class AvlSet: remove() |
| 2 | Class BstSet: contains() | Class AvlSet: contains() |
| 3 | Class BstSet: containsAll() | Class AvlSet: containsAll() |
| 4 | Class BstSet: addAll() | Class AvlSet: addAll() |
| 5 | Class BstSet: retainAll() | Class AvlSet: retainAll() |
| 6 | Class BstSet: remove() | Class java.util.TreeSet<E>: remove() |
| 7 | Class BstSet: contains() | Class java.util.TreeSet<E>: contains() |
| 8 | Class AvlSet: remove() | Class java.util.TreeSet<E>: remove() |
| 9 | Class AvlSet: contains() | Class java.util.TreeSet<E>: contains() |

Pastaba: Konkrečiu variantu gausite iš laboratorinių darbų dėstytojo.

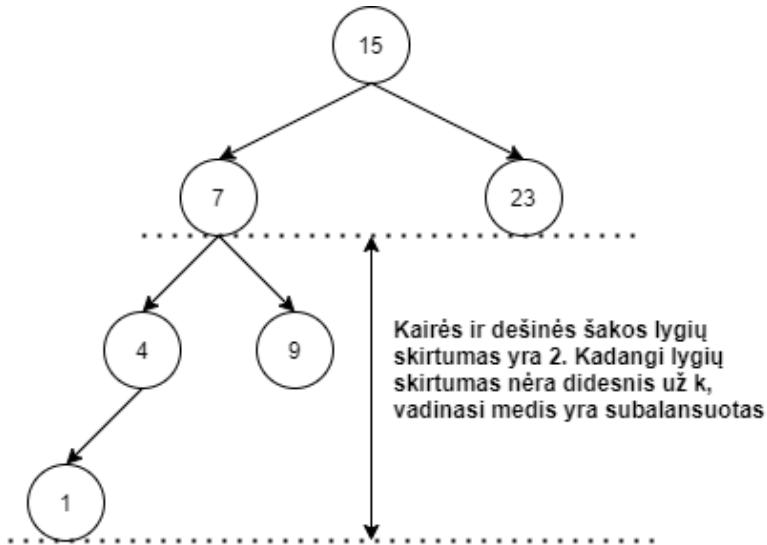
Laisva forma parenkite ataskaitą. Ataskaitoje būtinai turi būti:

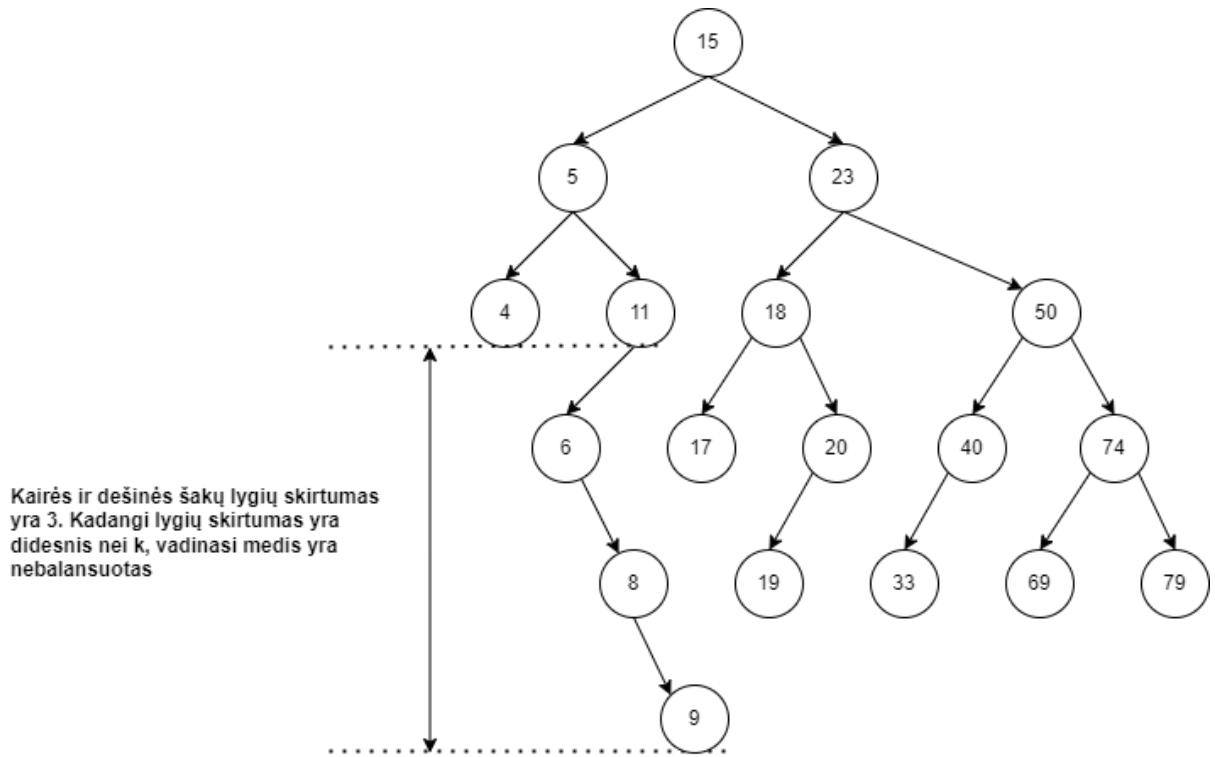
- Aprašyti tiriamieji metodai.
- Pateiktas tiriamujų metodų asymptotinis sudėtingumas.
- Aprašyta greitaveikos testavimo metodika (testavimo algoritmas).
- Aprašomi kompiuterio, su kuriuo buvo atlikti greitaveikos testai, pagrindiniai parametrai (procesoriaus charakteristikos, atminties kiekis ir pan.).
- Algoritmų/metodų vykdymo laiko priklausomybės nuo įvesties duomenų kiekių grafikas.
- Išvados. Išvadose laisva forma turėtų būti pateikti atsakymai į šiuos klausimus:
 - Ar eksperimentiškai nustatyta vykdymo laiko priklausomybė nuo įvesties dydžio atitinka teorinį algoritmo/metodo asymptotinį sudėtingumą. Jei ne, kodėl?
 - Kurie iš tirtujų metodų vykdymo laiko požiūriu yra geresni. Kodėl?

L2 neprivalomoji dalis

Išspręskite šiuos uždavinius:

1. (1 balas) Turime dvejetainį paieškos medį su jūsų pasirinktais duomenimis (medyje saugomas objektas turi realizuoti sąsają Comparable). Patikrinkite, ar medžio šakų balansas neviršija sveikojo skaičiaus k ($k \geq 0$). Pavyzdys, jeigu $k = 2$:



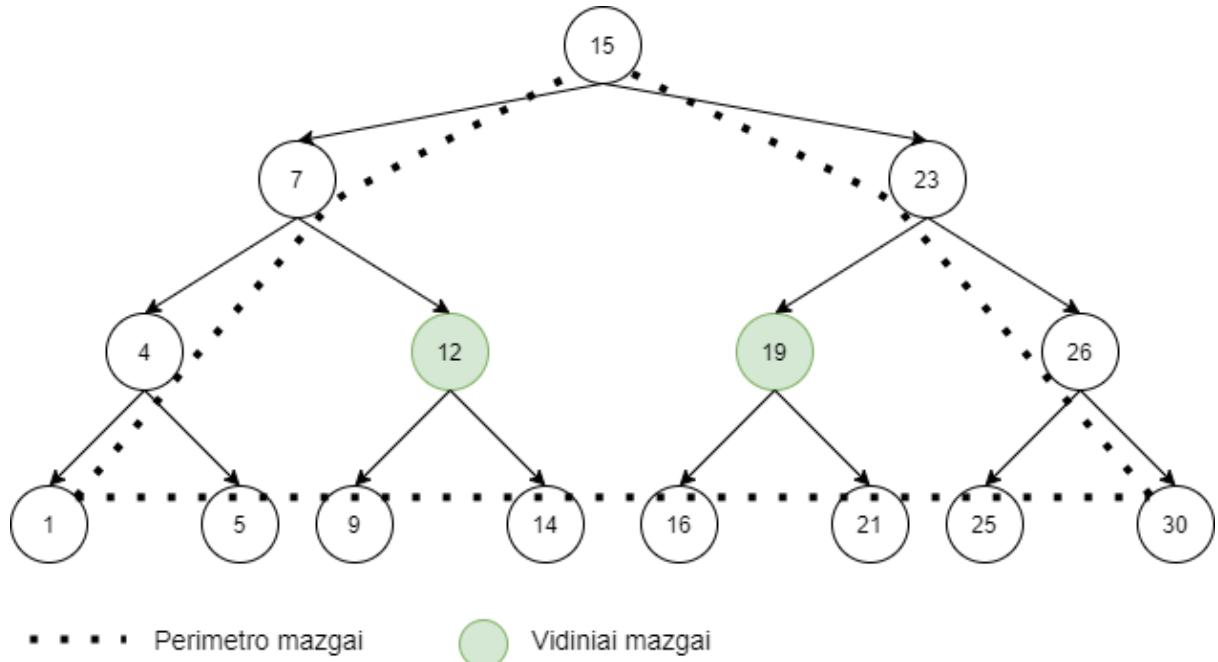


Ivesties duomenys:

- Skaičius k
- Dvejetainis paieškos medis

Parašykite algoritmą, kuris nustatyta, ar duotasis paieškos medis yra subalansuotas pagal parametrą k. Algoritmo asymptotinis sudėtingumas turi būti $O(n)$.

2. (1 balas) Turime pilnaji dvejetainį medį. Parašykite algoritmą, kuris rastų medžio vidinius mazgus, t.y mazgus, esančius medžio perimetro viduje. Pavyzdys:



LD2 gynimas

Gynimo metu dėstytojai gali studentų prašyti atlikti žemiau pateiktas užduotis ir atsakyti į žemiau pateiktus klausimus. Klausimų ir užduočių sąrašas nėra baigtinis - dėstytojai gynimo metu gali užduoti su tema susijusiu klausimu ar praktiniu užduočiu, kuriu nėra šiame sąraše.

Galimos praktinės užduotys iš privalomosios dalies

Realizuokite vieną ar kelis metodus iš šio sąrašo:

| Grąžinama reikšmė | Metodo aprašymas |
|-------------------|--|
| boolean | removeAll(BstSet<?> c) Removes from this set all of its elements that are contained in the specified collection. |
| E | ceiling(E e) Returns the least element in this set greater than or equal to the given element, or null if there is no such element. |
| E | floor(E e) Returns the greatest element in this set less than or equal to the given element, or null if there is no such element. |
| E | higher(E e) Returns the least element in this set strictly greater than the given element, or null if there is no such element. |
| E | last() Returns the last (highest) element currently in this set. |
| E | lower(E e) Returns the greatest element in this set strictly less than the given element, or null if there is no such element. |
| E | pollFirst() Retrieves and removes the first (lowest) element, or returns null if this set is empty. |
| E | pollLast() Retrieves and removes the last (highest) element, or returns null if this set is empty. |
| SortedSet<E> | copyOf(BstSet<? extends E> set) Returns a Set containing the elements of the given set |

Galimos praktinės užduotys iš neprivalomosios dalies

1. Parašykite algoritmą, kuris patikrintų, ar duotasis dvejetainis medis yra dvejetainis paieškos medis.
2. Parašykite metodą, kuris konvertuootuotų masyvą į balansuotą dvejetainį paieškos medį.

3. Parašykite metodą, kuris konvertuootuotų dvejetainį paieškos medžį į vienkryptį susietajį sąrašą.
4. Parašykite metodą, kuris leistų iteruoti per medžį išilgai jo lygiu.
5. Parašykite metodą, kuris leistų iteruoti per medžį vertikaliais sluoksniais nuo viršaus į apačią
6. Išspausdinkite dvejetainio paieškos medžio lapus.
7. Duoti du medžio mazgai, raskite ir išspausdinkite trumpiausią kelią tarp šių mazgų.
8. Raskite ir išspausdinkite visus dvejetainio paieškos medžio kelius nuo šaknies iki lapo.
9. Duota skaičius, nustatykite, ar dvejetainiame paieškos medyje egzistuoja tokis kelias, kurio mazgų raktų suma yra lygi duotajam skaičiui.
10. Perrašykite vieną iš pagrindinių medžio operacijų nenaudodami rekursijos.

Galimi teoriniai klausimai

1. Koks yra įvairių laboratoriiniame darbe realizuotų operacijų asymptotinis sudėtingumas?
2. Kokie yra dvejetainio paieškos medžio bei balansuoto dvejetainio paieškos medžio privalumai ir trūkumai?
3. Mokėkite paaškinti įvairius su programiniu kodu susijusius niuansus.