

## Trečioji užduotis

**Tikslas: įsisavinti paieškos dinaminėse duomenų aibėse procedūras**

1. Realizuoti dinaminės duomenų aibės operacijas *search* ir *insert*, raudonai-juodo medžio pagrindu.
2. Realizuoti dinaminės duomenų aibės operacijas *search* ir *delete*, raudonai-juodo medžio pagrindu.
3. Realizuoti dinaminės duomenų aibės operacijas *minimum*, *maximum*, *predecessor* ir *successor*, raudonai-juodo medžio pagrindu.
4. Realizuoti dviejų aibių *sajungos* ir *sankirtos* operacijas, kai kiekviena iš aibių yra išreikšta atskiru raudonai-juodu medžiu.
5. Realizuoti dviejų aibių *skirtumo* ir *simetrinio skirtumo* operacijas, kai kiekviena iš aibių yra išreikšta atskiru raudonai-juodu medžiu.
6. Realizuoti tikrinimo operacijas: *ar viena aibė yra kitos poaibis*, ir *ar dvi aibės yra lygios*, kai kiekviena iš aibių yra išreikšta raudonai-juodo medžio pagrindu.
7. Realizuoti dinaminės duomenų aibės operacijas *search* ir *insert*, 2-3-4 medžio pagrindu.
8. Realizuoti dinaminės duomenų aibės operacijas *search* ir *delete*, 2-3-4 medžio pagrindu.
9. Realizuoti dinaminės duomenų aibės operacijas *minimum*, *maximum*, *predecessor* ir *successor*, 2-3-4 medžio pagrindu.
10. Realizuoti dviejų aibių *sajungos* ir *sankirtos* operacijas, kai kiekviena iš aibių yra išreikšta atskiru 2-3-4 medžiu.
11. Realizuoti dviejų aibių *skirtumo* ir *simetrinio skirtumo* operacijas, kai kiekviena iš aibių yra išreikšta atskiru 2-3-4 medžiu.
12. Realizuoti tikrinimo operacijas: *ar viena aibė yra kitos poaibis*, ir *ar dvi aibės yra lygios*, kai kiekviena iš aibių yra išreikšta atskiru 2-3-4 medžiu.
13. Operacija *Join* dviems dinaminėms aibėms  $S_1$  ir  $S_2$  ir elementui  $x$  sudaro naują dinaminę aibę  $S$ , kurioje  $x_1 \leq x \leq x_2$ , kai  $x_1 \in S_1$  ir  $x_2 \in S_2$ . Realizuoti operaciją *Join* dviems raudonai-juodiems medžiams.
14. Operacija *Join* dviems dinaminėms aibėms  $S_1$  ir  $S_2$  ir elementui  $x$  sudaro naują dinaminę aibę  $S$ , kurioje  $x_1 \leq x \leq x_2$ , kai  $x_1 \in S_1$  ir  $x_2 \in S_2$ . Realizuoti operaciją *Join* dviems 2-3-4 medžiams.
15. Operacija *Split* dinaminei aibei  $S$  ir elementui  $x$  sudaro dvi dinamines aibes  $S_1$  ir  $S_2$ , kad  $x_1 \leq x \leq x_2$ , kai  $x_1 \in S_1$  ir  $x_2 \in S_2$ . Realizuoti operaciją *Split* raudonai-juodam medžiui.
16. Operacija *Split* dinaminei aibei  $S$  ir elementui  $x$  sudaro dvi dinamines aibes  $S_1$  ir  $S_2$ , kad  $x_1 \leq x \leq x_2$ , kai  $x_1 \in S_1$  ir  $x_2 \in S_2$ . Realizuoti operaciją *Split* 2-3-4 medžiui.
17. Ranginės statistikos (order statistics) medis yra raudonai-juodas medis, kurio kiekvienoje viršūnėje saugoma papildoma informacija – pomedžio, prasidedančioje toje viršūnėje viršūnių skaičius. Realizuoti algoritmą, kuris tokioje struktūroje randa  $k$ -ąjį mažiausią elementą.
18. Ranginės statistikos (order statistics) medis yra raudonai-juodas medis, kurio kiekvienoje viršūnėje saugoma papildoma informacija – pomedžio, prasidedančioje toje viršūnėje viršūnių skaičius. Realizuoti algoritmą, kuris tokioje struktūroje randa  $k$  mažiausių elementų.
19. Ranginės statistikos (order statistics) medis yra raudonai-juodas medis, kurio kiekvienoje viršūnėje saugoma papildoma informacija – pomedžio,

prasidedančioje toje viršūnėje viršūnių skaičius. Realizuoti dinaminės duomenų aibės operacijas *minimum*, *maximum*, *predecessor* ir *successor* ranginės statistikos medžio pagrindu.

20. Ranginės statistikos (order statistics) medis yra raudonai-juodas medis, kurio kiekvienoje viršūnėje saugoma papildoma informacija – pomedžio, prasidedančioje toje viršūnėje viršūnių skaičius. Realizuoti dinaminės duomenų aibės operacijas *insert* ir *search* ranginės statistikos medžio pagrindu.
21. Ranginės statistikos (order statistics) medis yra raudonai-juodas medis, kurio kiekvienoje viršūnėje saugoma papildoma informacija – pomedžio, prasidedančioje toje viršūnėje viršūnių skaičius. Ranginės statistikos medžio pagrindu realizuoti algoritmą, kuris duotiems skaičiams  $a$  ir  $b$  suranda visus  $x$ , tokius kad  $a < x < b$ .
22. Intervalų medis yra raudonai-juodas medis, kurio kiekvienoje viršūnėje saugomi du skaičiai – skaičių intervalas (pirmas skaičius, visuomet mažesnis už antrą). Realizuoti algoritmą, surandantį visus intervalus, kuriuose yra duotas skaičius.
23. Realizuoti dinaminės duomenų aibės operacijas *minimum*, *maximum*, *predecessor* ir *successor*, kai dinaminės aibės elementai yra intervalai, ir ji išreikšta intervalų medžio pagrindu.
24. Intervalų medis yra raudonai-juodas medis, kurio kiekvienoje viršūnėje saugomi du skaičiai – skaičių intervalas (pirmas skaičius, visuomet mažesnis už antrą). Realizuoti algoritmą, surandantį visus intervalus, susikertančius su duotu intervalu, ir bent vieną intervalą, nesikertantį su duotu intervalu.
25. Intervalų medis yra raudonai-juodas medis, kurio kiekvienoje viršūnėje saugomi du skaičiai – skaičių intervalas (pirmas skaičius, visuomet mažesnis už antrą). Realizuoti algoritmą, surandantį visus intervalus, kuriuose yra duotas skaičius.
26. Sukurti ir realizuoti algoritmą, kuris duotai skaičių aibei  $x_1, x_2, x_3, \dots, x_n$  suranda mažiausią aibę vienetinio ilgio uždarų intervalų, kurie apima šiuos skaičius (pastaba: galima pasinaudoti Huffmano kodo įdėjomis)
27. Automobilio degalų pilno bako užtenka nuvažiuoti  $n$  kilometrų. Studentas turi iš viso nuvažiuoti  $m$  kilometrų, o jo kelyje yra išdėstytos degalinės, tarp kurių atstumai yra žinomi. Sukurti algoritmą ir parašyti programą, kuri nurodo kuriomis degalinėmis pasinaudoti, kad pildyti baką reikėtų mažiausią galimą kiekį kartų.
28. Realizuoti dinaminės duomenų aibės operacijas *search* ir *insert*, B-medžio pagrindu, kai  $M=7$ .
29. Realizuoti dinaminės duomenų aibės operacijas *search* ir *delete*, B-medžio pagrindu, kai  $M=8$ .
30. Realizuoti dinaminės duomenų aibės operacijas *minimum*, *maximum*, *predecessor* ir *successor*, B-medžio pagrindu, kai  $M=9$ .
31. Realizuoti dviejų aibių *sajungos* ir *sankirtos* operacijas, kai kiekviena iš aibių yra išreikšta atskiru B-medžiu,  $M=7$ .
32. Realizuoti dviejų aibių *skirtumo* ir *simetrinio skirtumo* operacijas, kai kiekviena iš aibių yra išreikšta atskiru B-medžiu,  $M=9$ .
33. Realizuoti tikrinimo operacijas: *ar viena aibė yra kitos poaibis*, ir *ar dvi aibės yra lygios*, kai kiekviena iš aibių yra išreikšta B-medžio pagrindu, kai  $M=6$ .
34. Parašyti programą, kuri skaičiuoja vidutinį trie-medžio viršūnių skaičių, ir vidutinį medžio aukštį, įterpiant  $N$  atsitiktinių skaičių, kai  $N$  yra 1000 ir 10000.