|  |
| --- |
| **Teme:** **Enorazsežne tabele - utrjevanje**   * delo s celotno tabelo, * linearni postopki: traversing, iskanje |

|  |
| --- |
| **! – skušajte že spisano, če se le da, večkrat uporabiti.** |

**Naloga 1**

Spišite metodo razreda Naloga11\_1, ki bo z argumentom podano tabelo razbila v 2. Prva rezultativna tabela naj vsebuje zgolj lihe, druga zgolj sode vrednosti. Privzemite, da je morebitna vrednost 0 soda. Po procesu razbijanja, naj bo izveden tudi izpis vsebine vseh treh tabel.

Namig:

int[] vh1 = {1,2,3,4,5,6,7,7,7,7,8,9};

int[] vh2 = new int[60];

int[] rezLihe = null; // naredimo, ko vemo, koliko lihih je v vhodni tabeli

int[] rezSode = null;

morebitni klici : razbij1(vh1);

razbij1(vh2); // rezultate vedno prepisuje(ta) v razredne lastnosti rezLihe in rezSode

razbij2(vh1,razLihe,razSode); // rezultati v argumentih

**Naloga 2**

Spišite metodo razreda z imenom truncToLengh/2, ki ji podate tabelo števil (ali poljubnih celih ali necelih), postopek metode pa tabelo skrajša na dolžino, ki je podana z drugim argumentom metode.

**Naloga 3**

Metoda nastaviZacetneKoncne/3 v tabeli, podani s prvim argumentom nastavi prvih in zadnjih x elementov na vrednost v. X je vrednost drugega, v pa tretjega argumenta metode. Metoda naj vsebinsko spremenjeno tabelo vrne z običajnim mehanizmom vračanja. Spišite tudi test klica metode, ki po izvršenem klicu izpiše obe tabeli; originalno in spremenjeno. (in ah, samostojno izvedite odločitev, kako boste servisirali klice, kjer je tabela krajša od 2\*x)

**Naloga 4**

Spišite razredno metodo, ki bo ustvarila in vrnila tabelo celih števil izbrane dolžine:

int[] t1 = ustvariTabInt(33,true); // ustvari tabelo dolžine 33, vrednosti nepadajoče urejene

int[] t2 = ustvariTabInt(1333,false); // 1333 elementov, neurejena (pseudo-naključno urejena)

**Naloga 5**

Javanski program ustvari tabelo dolžine k\*100. K naj bo naključno število iz intervala [2,9]. Tabelo naj napolni s pomočjo metode predhodne naloge. Nato naj zaporedno izpiše vsa nepadajoče urejena podzaporedja osnovnega zaporedja, zapisanega v tabeli, vsako izmed podzaporedij naj izpiše v lastni vrstici.

**Naloga 6**

1. Javanski program Naloga11\_6 vsebuje 3 javne razredne metode; 'driver' ter metodi prestejZap/0, poisciNajd/0. PrestejZap/0 v razredno definirani tabeli celih števil prešteje in vrne število ugotovljenih urejenih podzaporedij, PoisciNajd/0 pa v teistem poišče in vrne pozicijo najdaljšega urejenega podzaporedja.
2. Dodajte razredu splošnejši metodi prestejZap in poisciNajd, ki bosta svojo funkcijo znali opraviti na poljubnem zaporedju celih števil podanim z argumentom metode.

**Naloga 7**

Spišite razredno metodo, ki bo ugotovila, ali je zaporedje v podani tabeli nepadajoče urejeno ali ne.

Spišite še metodo, ki na morebiti istem zaporedju ugotovi nenaraščajočo urejenost.

**Naloga 8**

Javanski program ob pričetku izvajanja generira dve (enako) urejeni tabeli. Postopek v nadaljevanju s pomočjo metode zlij/2 obe (z argumentom metodi podani) tabeli prepiše v eno samo, ki je tudi sama urejena na enak način, kot sta urejeni podani tabeli. Spisan postopek naj bo linearne narave. (poskusite s sočasnim zaporednim branjem obeh tabel)

**Naloga 9**

Spišite metodo razbij2/4, ki pri klicu razbij2(tabOsno,4,rezt1,rezt2) razbije tabelo tabOsno v dve tako, da prve 4 elemente prepiše zaporedno v tabelo rez1,naslednje 4 na rez2, naslednje 4 spet na rez1 . Pazite na dolžine, upoštevajte tudi, da dolžine tabOsno ni nujno mnogokratnih velikosti drugega argumenta pri klicu.