|  |
| --- |
| ***Utrjevanje:***   * Osnove OOP |

1. Dobili smo naročilo za razred, s katerim bi lahko sledili drevesnim vrstam v gozdu. Seveda bi želeli imeti tudi njegovo realizacijo oz. razred, ki bo demonstriral delovanje.

Torej, razred Drevo naj bi vseboval podatke o vrsti drevesa (bor, smreka,…), letnici (kdaj je vsajeno), obsegu in višini drevesa. Vse lastnosti naj bodo zaščitene (privatne, private).

Razred naj vsebuje tudi ustrezni(e) konstruktor(je) in:

1. metodo, ki izračuna in vrne, kako hitro raste posamezno drevo (povprečni letni prirast v cm);
2. metodo, ki preveri, ali dve drevesi iste drevesne vrste rasteta enako hitro oz. imata enako povprečni letni prirast.

* Dodajte razredno metodo, ki bo omogočila računanje največje razlike v obsegu debla med enako starimi drevesi iste drevesne vrste
* Dodajte tudi razredno metodo, ki bo omogočila računanje največje razlike s smislu »Kakšna je maksimalna razlika v obsegu debla med enako starimi drevesi različnih drevesnih vrst«.

Po potrebi lahko dodate še kakšno objektno metodo.

* Ustvarite 4 objekte razreda Drevo – podatke podaj v konstruktorju, kot argument(e) konstruktorja in s pomočjo nastavljalcev(setterjev). Izpišite povprečno rast posameznega drevesa (objekta).
* Preverite, koliko zasajenih smrek raste enako hitro kot prva smreka in to izpiši.
* Izračunajte in izpiši največjo razliko v obsegu debla med drevesi iste vrste in različnih vrst.
* Narišite ustrezen razredni diagram.

1. Deklarirajte razred Ulomek, ki vsebuje zaščitene (private) lastnosti ***stevec*** in ***imenovalec*** (lastnosti naj bosta celi števili. Razred naj vsebuje 3 konstruktorje:
   1. konstruktor, ki nima parametrov – default vrednosti sta 1, 1;
   2. konstruktor z enim argumentom (vrednost stevca) in postavi imenovalec na 1.
   3. konstruktor, ki dobi dva parametra, s katerima nastavi vrednosti obeh lastnosti.

Razred vsebuje še naslednje objektne metode :

1. metodo, ki preveri, ali je možno pokrajšati ulomek in to tudi naredi (če je možno seveda),
2. metodo enakaUlomka, ki preveri, ali sta dva ulomka enaka in to tudi vrne;
3. metodo, ki vrne niz znakov, ki predstavlja ulomek. Metoda naj bo prekrivanje(overrun) metode toString in naj vrne niz kot za primer: če je števec – 12 in imenovalec 5, naj vrne "-12 / 5";
4. metodo, ki sešteje dva ulomka in vrne novo nastali ulomek.

Kot demonstracijo delovanja ustvarite tri ulomke (uporabite različne konstruktorje), preverite ali je katerikoli ulomek možno okrajšati (in ga v tem primeru tudi okrajšajte). Izpišite, koliko ulomkov je enakih in tudi, kateri ulomki so enaki. Na koncu seštejte vse tri ulomke in seštet ulomek tudi izpiši (toString).

Po potrebi lahko dodate še kakšno objektno metodo. PAZI! Preverjaj vrednost imenovalca!

Narišite tudi ustrezen razredni diagram.