**Vaje iz preobloženih operatorjev, razrednih šablon in statičnih članov**

1. Napišite program v C++, ki deklarira razred Valj s privatnimi celoštevilčnimi lastnostmi r in v.

Razred na vsebuje privzeti (obe lastnosti postavi na 0) konstruktor, kopirni konstruktor, destruktor, metodo za branje obeh lastnosti s tipkovnice (ta naj ponovi branje, v koliko uporabnik vnese števila <0) , metodi za izračun površine in prostornine valja. Razred naj vsebuje še preobložene operatorje: != (vrne logično 1, če sta prostornini valjev različni), operator += (poveča obe lastnosti valja za velikost vhodnega parametra n) ter operator -. Slednji naj deluje tako, da naj najprej preveri radij obeh valjev. V kolikor je radij različen, naj vrne objekt razreda Valj, ki ima obe lastnosti enaki -1. Če sta radija enaka, pa naj operator – vrne objekt razreda Valj, katerega radij je enak radiju trenutnega objekta, višina pa naj se izračuna kot absolutna vrednost razlike višin obeh valjev. V glavnem programu omogočite vnos lastnosti objektov a in b. Za objekt a kličite operator a+=3, ter izračunajte razliko med objema objektoma. Nato še izvedite primerjavo med objektoma a in b. Če sta objekta enaka, naj program izpiše ustrezno sporočilo, če pa ne, pa naj program izpiše kakšna je razlika med površinama obeh objektov. Ob koncu programa še sprostite oba objekta iz pomnilnika.

1. Nalogo pod točko 1. še spremenite tako, da bo razred Valj uporabljal generični tip T za lastnosti radij in visina ter v glavnem programu določite, da naj bo tip T tipa int in nato še tipa short.
2. Napišite program v C++, ki deklarira razred *Tocka* z lastnostmi x, y in oblika. Lastnost oblika naj bo znakovnega tipa, lastnosti x in y pa naj bosta kratki celi števili. Vse lastnosti naj bodo zasebne (*private*). Razred naj vsebuje še metodo za branje obeh lastnosti, ki ne sme dopustiti vnos negativnih vrednosti, privzeti ter pretvorbeni konstruktor, preobloženi operator inkrementa (++), preobloženi operator \*= ter preobloženi operator enačaj (a==b). Ta operator primerja med seboj dva objekta. Objekta sta enaka, kadar sta obe njuni lastnosti x in y enaki. Operator inkrementa (++) poveča lastnost x za 1, operator \*= pa naj množi lastnosti x in y za toliko, kolikor znaša vrednost vhodnega parametra. V glavnem programu ustvarite objekt a s pomočjo privzetega konstruktorja ter objekt b s pomočjo pretvorbenega konstruktorja (pretvorite v tip float). V glavnem programu omogočite vnos lastnosti obeh objektov a in b. Za objekt a kličite operator ++, za objekt b pa operator b+=3. Če sta objekta enaka, naj program izpiše ustrezno sporočilo, če pa ne, pa naj program izpiše kakšna je razdalja med njima (d=sqrt(pow(x1-x0,2)+pow(y1-y0,2));). Za računanje razdalje uporabite vmesnike (interface).
3. Nalogo pod točko 2. še spremenite tako, da bo razred Tocka uporabljal generični tip T1 za koordinato x ter T2 za koordinato y ter v glavnem programu določite, da naj bo najprej tip T1 enak tipu short, T2 pa tipu long, nato pa naj bo tip T1 enak tipu float, tip T2 pa tipu double.
4. Napišite program v C++, ki deklarira razred *BancniRacun* s privatno tabelo pologi\_dvigi, ki naj vsebuje največ 100 realnih števil. Razred naj vsebuje tudi realno spremenljivko (gre za statičnega člana, ki ga na začetku postavimo na vrednost 0), privzeti konstruktor (vse elemente tabele pologi\_dvigi postavi na 0), ter destruktor. Razred naj vsebuje metodo za branje, ki bere pologe in dvige z računa ter jih shranjuje v tabelo artikli, dokler uporabnik ne vnese števila 0. Metoda naj dovoljuje vnos tudi negativnih vrednosti (to so dvigi), vendar naj zaključi vnos takoj, ko pridemo v negativno stanje računa. Metoda za izpis naj izpiše vse pologe in dvige z računa ter trenutno stanje na računu. Razred naj vsebuje še preobložena operatorje +, -- ter ==. Operator == naj vrne vrednost 1, če je skupna vsota dvigov in pologov (stanje) prvega računa enako stanju drugega računa. Operator + naj vsakemu pologu (pozitivna števila) na računu prišteje n % obresti, pri čemer naj bo n vhodni parameter funkcije. Operator – naj odstrani zadnji vnos (polog ali dvig) z računa tako, da njegovo vrednost postavi na 0. V glavnem programu ustvarite objekta rac1 in rac2 s pomočjo privzetega konstruktorja ter kličite metodi za vnos in izpis objektov rac1 in rac2. Nato za objekt rac1 kličitte operator +, za objekt rac2 pa operator --. S pomočjo operatorja == preverite in izpišite, če sta objekta enaka. Program naj s pomočjo statične spremenljivke še izračuna razliko med obema objektoma rac1 in rac2. Na koncu še sprostite oba objekta iz pomnilnika.
5. Napišite program v C++, ki deklarira razred *Nizi* s privatno lastnostjo - tabelo znaki, ki naj bo znakovnega tipa in naj vsebuje največ 100 znakov. Razred naj vsebuje metodo za vnos lastnosti znaki (ta naj dovoljuje tudi vnos presledkov) in metodo za izpis, ki izpiše tabelo znaki in tudi koliko črk (velike ali male črke angleške abecede) ter koliko preostalih znakov je vnesenih v tabeli znaki. Razred naj vsebuje metodo, ki shrani tabelo znaki na konec datoteke besede.txt, v kolikor pa datoteka ne obstaja, pa naj jo metoda naredi. Razredu naj bo dodana metoda, ki bere n-ti zaporedni niz iz datoteke besede.txt v tabelo znaki. Razred naj vsebuje tudi preobložene operatorje +, -- ter ==. Operator == naj vrne vrednost 1, če je sta tabeli znaki dveh objektov enaki. Operator + ustvari nov objekt razreda Nizi, ki ga dobimo tako, da združimo (zlepimo) tabelo znakov prvega objekta in tabelo znakov drugega objekta, a le če še nismo presegli zgornje meje števila znakov (100). Operator -- naj odstrani zadnjo črko v tabeli znaki (prestavimo kontrolni znak \0 za eno mesto nazaj.). V glavnem programu ustvarite objekte beseda1, beseda2 in beseda 3. Za objekt beseda1 kliči metodo za vnos s tipkovnice, za objekt beseda2 pa kliči metodo za branje 3. zaporednega niza iz datoteke besede.txt. Nato za objekt beseda2 kliči operator --, objekt beseda3 pa naj postane enak vsoti (operator +) objektov beseda1 in beseda2. S pomočjo operatorja == izvedite primerjavo med objektoma beseda2 in beseda3. Nato shranite vrednosti objekta 3 na datoteko (klic metode za shranjevanje niza na datoteko) ter izpišite vrednosti vseh objektov na ekran. S pomočjo statične spremenljivke dolzinaSkupaj še izpišite, koliko je znašala dolžina vseh nizov (lastnost - tabela znaki) objektov skupaj.
6. Napišite program v C++, ki bo deklariral razred MatrixReloaded, ki naj vsebuje privatno dostopno tabelo celih števil dimenzije 3x3. V razredu implementirajte vse preobložitve naslednjih operatorjev:
7. Operator +

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | -4 | 6 |  | -7 | 10 | 2 |  | -5 | 6 | 8 |
| 3 | 8 | 1 | **+** | 3 | -1 | 9 | **=** | 6 | 7 | 8 |
| 4 | 5 | -3 |  | -4 | 5 | 3 |  | 0 | 10 | 7 |

1. Operator -

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | -4 | 6 |  | -7 | 10 | 2 |  | 9 | -14 | 4 |
| 3 | 8 | 1 | **-** | 3 | -1 | 9 | **=** | 0 | 9 | -8 |
| 4 | 5 | -3 |  | -4 | 5 | 3 |  | 8 | 0 | -6 |

1. Operator \*=n (primer za n=3)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | -4 | 6 |  |  |  | 6 | 12 | 18 |
| 3 | 8 | 1 | **\*** | 3 | **=** | 9 | 24 | 3 |
| 4 | 5 | -3 |  |  |  | 12 | 15 | -9 |

1. Operator +=n (primer za n=3)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | -4 | 6 |  |  |  | 5 | -4 | 6 |
| 3 | 8 | 1 | **+** | 3 | **=** | 3 | 11 | 1 |
| 4 | 5 | -3 |  |  |  | 4 | 5 | 0 |

1. Operator -=n (primer za n=3)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | -4 | 6 |  |  |  | 2 | -4 | 3 |
| 3 | 8 | 1 | **-** | 3 | **=** | 3 | 5 | 1 |
| 4 | 5 | -3 |  |  |  | 1 | 5 | -3 |

V glavnem programu deklarajte dva objekta razreda MatrixReloaded ter kličite vse njune metode. Ob koncu programa še sprostite objekta iz pomnilnika.

1. Napišite program v C++, ki deklarira imenski prostor *PRavokotnik z* lastnostnima a in b, ki naj bosta celoštevilčnega podatkovnega tipa. Imenski prostor naj vsebuje še metode za branje lastnosti a in b (ta ne sme dovoliti vnos negativnih vrednosti), za izpis obeh lastnosti, metodo za izračun ploščine in metodo za izračun obsega pravokotnika. V glavnem programu kličite vse metode imenskega prostora za dva pravokotnika ter na koncu še izpišite, kateri pravokotnik je imel manjšo ploščino.