

Napišite sučelje i implementaciju za klase **Cetverokut**, **Trapez**, **Pravokutnik**, **Kvadrat**. Sljedeći dijagram opisuje relacije među klasama. Sučelja za klase spremite u datoteku **cetverokut.h**, a implementacije u datoteku **cetverokut.cpp**.



Klasa Cetverokut

Ova klasa opisuje četverokut. Svaki je četverokut određen sa 4 vrha u ravni.

- `Cetverokut(točka x, točka y, točka z, točka w)` – konstruktor kojem zadajemo 4 točke u ravni koje određuju četverokut
- `double površina()` - vraća površinu četverokuta
- `double opseg()` - vraća opseg četverokuta
- `std::set<double> kutovi()` - vraća kutove četverokuta (u stupnjevima)
- `std::string tipCetverokuta()` - vraća najuži tip četverokuta kojem pripada. Točnije, vraća jednu od vrijednosti `std::string("cetverokut")`, `std::string("trapez")`, `std::string("pravokutnik")`, `std::string("kvadrat")`
- `bool ispravanTip()` - vraća `true` ako je dan četverokut (vrhovi tvore četverokut), inače `false`

Klasa Trapez

Ova klasa opisuje trapez. Svaki je trapez određen sa 4 vrha u ravni ili 4 stranice. Osim što nasljeđuje funkcije klase `Cetverokut` definira/redefinira još minimalno ove funkcije:

- `Trapez(double a, double b, double c, double d)` – konstruktor kojem zadajemo 4 stranice trapeza. Podrazumijeva se da su `a` i `c` paralelne.
- `Trapez(točka x, točka y, točka z, točka w)` – konstruktor kojem zadajemo 4 točke u ravni koje određuju trapez
- `bool ispravanTip()` - vraća `true` ako je dan trapez (stranice/vrhovi tvore trapez), inače `false`

Klasa Pravokutnik

Ova klasa opisuje pravokutnik. Svaki je pravokutnik određen sa 4 vrha u ravnini ili 4 stranice. Osim što nasljeđuje funkcije klase Cetverokut definira/redefinira još minimalno ove funkcije:

- Pravokutnik(double a, double b) – konstruktor kojem zadajemo 2 stranice pravokutnika. Podrazumijeva se da su a i b okomite.
- Pravokutnik(točka x, točka y, točka z, točka w) – konstruktor kojem zadajemo 4 točke u ravnini koje određuju pravokutnik
- bool ispravanTip() - vraća true ako je dan pravokutnik (stranice/vrhovi tvore pravokutnik), inače false

Klasa Kvadrat

Ova klasa opisuje kvadrat. Svaki je kvadrat određen sa 4 vrha u ravnini ili 4 stranice (iste). Osim što nasljeđuje funkcije klase Cetverokut definira/redefinira još minimalno ove funkcije:

- Kvadrat(double a) – konstruktor kojem zadajemo 1 stranicu kvadrata.
- Kvadrat(točka x, točka y, točka z, točka w) – konstruktor kojem zadajemo 4 točke u ravnini koje određuju kvadrat
- bool ispravanTip() - vraća true ako je dan kvadrat (stranice/vrhovi tvore kvadrat), inače false

Napomene:

- točka = typedef std::pair<double,double> točka
- kod konstruktora kojem se zadaju 4 točke, točke navodimo u smjeru kazaljke na satu.

Opće napomene

- Struktura, funkcije i datoteke koje šalžete moraju se zvati točno onako kako je zadano u zadatku. Pazite na mala i velika slova!
- Trebate poslati samo sučelje i implementaciju. U datotekama koje šalžete ne smije se nalaziti funkcija main()!
- nijedna funkcija ne smije ništa učitavati s tipkovnice ili neke datoteke, niti išta ispisivati na ekran ili u neku datoteku.
- Svaki od main-ova pomoću kojih testiramo ispravnost vašeg programa neće pozivati sve gore navedene funkcije. Stoga, ako neku od funkcija ne znate napisati ipak možete dobiti koji bod (u tom slučaju tu funkciju nemojte navesti niti u .h niti u .cpp datoteci ili napravite neku trivijalnu implementaciju).

Ispravnost implementacija koje napišete bit će provjerena tako da ćemo mi napisati razne klijentske programe koji će deklarirati nekoliko varijabli zadane strukture, i na njima pozivati funkcije koje ste trebali napisati. Ako se poslani programi ne budu uspješno povezivali (linkali) s našim klijentskim programima, smatrat će se neispravnima.

Neki klijentski programi provjeravati će samo neke jednostavnije funkcije, dok će neki provjeravati sve funkcije koje trebate napisati. Provjera je potpuno automatska, tako da je od presudne važnosti da se pridržavate specifikacije. Nepridržavanje lako može uzrokovati osvojenih 0 bodova iz zadaće!

Naravno, za provjeru radi li implementacija prije nego što je pošaljete, preporučuje se da je testirate pomoću nekog klijentskog programa. No taj klijentski program ne šaljete!

Mole se studenti da ne kontaktiraju asistenta Ugrinu preko maila već da sva pitanja vezana uz zadatak postavljaju na Forumu pod temom "(2013) 3. zadaca Cetverokut".