

# Računarski praktikum 1

## Četvrta zadaća

### Treća grupa

#### Surd

*Surd* je izraz oblika  $a+b\sqrt{c}$ , gdje su  $a$ ,  $b$  i  $c$  cijeli brojevi, a  $\sqrt{c}$  oznaka za drugi korijen ( $c$  može biti i negativan; kompleksni brojevi su specijalni slučaj surdova za  $c=-1$ ). Implementirajte klasu *Surd*, čije vrijednosti predstavljaju surdove kojima su  $a$ ,  $b$  i  $c$  tipa *int*, te još jednu specijalnu vrijednost *Surd::NaS* ("*not a surd*"), koja služi za signaliziranje grešaka. *NaS* se ne smije moći konstruirati niti mijenjati izvana -- može se samo dobiti provođenjem neke operacije čiji rezultat nije surd.

Surdovi se trebaju automatski pojednostavljivati, npr.  $2+3\sqrt{8}=2+6\sqrt{2}$ , ili  $5-2\sqrt{36}=5-2\cdot 6=-7$ . (Uputa: dijelite  $c$  kvadratima prirodnih brojeva dok možete.)

Klasa mora sadržavati:

- konstruktor *Surd(int x)*, koji stvara surd jednak  $x$  (cijeli brojevi su surdovi, oblika  $x+0\sqrt{0}$ ). Defaultni konstruktor stvara surd 0.
- konstruktor *Surd(int x,int y,int r)*, koji stvara surd  $x+y\sqrt{r}$ .
- operator *cast* u *double*, koji vraća približnu vrijednost surda, ako je  $c$  nenegativan. Za negativne  $c$  vraća 0.
- operator *cout<<Surd*, koji ispisuje surd na *cout*. *NaS* se ispisuje točno s ta tri znaka ("*NaS*"), dok se surdovi ispisuju u skladu s uobičajenom matematičkom notacijom, koristeći znak *@* za drugi korijen. Na primjer, *Surd(-3,2,4)* se ispisuje kao 1, *Surd(4,-1,2)* kao 4-@2, *Surd(-2,3,3)* kao -2+3@3, a *Surd(0,2,18)* kao 6@2. Pazite na predznake!
- operatore *++* (prefiksni i postfiksni), koji vraćaju "cijeli i necijeli dio" surda: *++Surd* vraća  $a$ , a *Surd++* vraća  $b$  (naravno, nakon automatskog pojednostavljivanja). Omogućite mijenjanje  $a$  i  $b$  pomoću tih operatora (ako operand nije *const*): na primjer, *Surd s(4,-1,2);s++=6;*
- operatore *==* i *<*, koji služe za uspoređivanje dva surda (vraćaju *bool*).
- binarne operacije *+*, *-* i *\**, koje vraćaju zbroj, razliku odnosno produkt dva surda, ako je to surd. Ako nije, vraćaju *NaS*. Na primjer,  $(1+\sqrt{2})\cdot(3-2\sqrt{2})=-1+\sqrt{2}$ , dok  $(1+\sqrt{2})-(1+\sqrt{3})=\text{NaS}$ . Uputa: ako ste pojednostavili surdove do kraja, rezultat će biti surd ako i samo ako je jedan od operandi cijeli broj, ili oba operanda imaju isti " $c$ " pod korijenom.
- unarni operatori: *-*, *~* (konjugirani surd,  $a-b\sqrt{c}$ ), te *\** koji vraća normu surda. Norma  $x$  se definira kao  $x\cdot\sim x$ , i uvijek je cijeli broj.
- operator *Surd^int n*, koji vraća potenciju surda za  $n\geq 0$ . Inače vraća 0.

### Opće napomene

- Operatori bi trebali raditi s const objektima gdje god je to moguće i gdje ima smisla. Možda ćete morati imati dvije varijante istog operatora, kako biste osigurali mogućnost rada s konstantnim i nekonstantnim objektima.
- Operatore smijete pisati kao članske ili kao globalne funkcije (osim onih koji moraju biti jednog od ta dva tipa). Ipak, uzmite u obzir da se binarni operatori trebaju moći izvršavati kako treba i u slučaju kad lijevi operand “nije pametan” (a desni jest).
- Omogućite ulančavanje operatora gdje god možete. To se posebno odnosi na ispis na cout.