

8. Ako je zbir neke tri stranice pravougaonika  $m$ , a zbir neke druge tri stranice istog pravougaonika  $n$ , napisati program kojim se određuje obim i površina tog pravougaonika. Prva linija standardnog ulaza sadrži realan broj  $m$  koji predstavlja zbir neke tri stranice pravougaonika, druga linija sadrži realan broj  $n$  koji predstavlja zbir neke druge tri stranice pravougaonika. U prvoj liniji standardnog izlaza prikazati na dve decimale obim pravougaonika, a u drugoj liniji površinu pravougaonika takođe na dve decimale.

Sledeći C++ program sadrži više grešaka. Ispraviti sve greške.

```
include iostream
int main() { double n, m;
cin >> n >> m; a = (2 * n - m) / 3; b = (2 * m) / 3; O = 2 * (a + b); P = a * b;
cout << O << P << endl;}
```

## REŠENJE

### *Analiza*

Obeležimo stranice pravougaonika sa  $a$  i  $b$ .

Tada je obim pravougaonika  $2*a + 2*b$ , a površina je  $a*b$ .

Ako je prvi zbir tri stranice pravougaonika  $2*a + b$ , onda je zbir druge tri stranice  $a + 2*b$ .

Tako dobijamo dve jednačine iz kojih možemo odrediti  $a$  i  $b$ .

### *Smernice za algoritam*

#### **I način**

Jedan način da se dođe do rešenja je da se reši sistem jednačina (čime se dobija da je

$$a = (2*n - m) / 3$$

$$b = (2*m - n) / 3$$

```
#include <iostream>
#include<iomanip>
using namespace std;
```

```
int main() {
double n, m;
cin >> n >> m;
double a = (2 * n - m) / 3.0;
double b = (2 * m - n) / 3.0;
double O = 2 * (a + b);
double P = a * b;
cout << fixed << showpoint << setprecision(2) << O << endl << P << endl;
return 0;
}
```

## II način

Ali, do rešenja se može doći i bez toga.

Sabiranjem jednačina

$$2*a + b = n,$$

$$a + 2 * b = m$$

dobijamo da zbir stranica a i b (označavaćemo ga kao *zbirStranica*) iznosi  $(n+m)/3$ . Obim se izračunava kao dvostruki zbir stranica. Za određivanje površine potrebno je odrediti proizvod stranica a i b. Stranicu a dobijamo kada od datog zbira tri stranice n oduzmemo zbir stranica a i b ( $n - \text{zbirStranica}$ ). Slično dobijamo i stranicu b kao  $m - \text{zbirStranica}$ .

```
#include <iostream>
```

```
#include<iomanip>
```

```
using namespace std;
```

```
int main() {
```

```
    double n, m;
```

```
    cin >> n >> m;
```

```
    double zbirStranica = (m + n) / 3;
```

```
    double O = 2 * zbirStranica;
```

```
    double a = m - zbirStranica;
```

```
    double b = n - zbirStranica;
```

```
    double P = a * b;
```

```
    cout << fixed << showpoint << setprecision(2) << O << endl << P << endl;
```

```
    return 0;
```

```
}
```

9. Tri pastira Jovan, Dule i Stanoje broje stado od najviše 120 ovaca na kraju svakog dana, na prste desne ruke. Broje sva trojica na sledeći način:

Jovan broji jednu po jednu ovcu na prste desne ruke. Svaki put kad izbroji pet ovaca on kaže Duletu da izbroji još jedan prst na svojoj desnoj ruci, a Jovan krene brojanje od početka.

Dule broji jedan po jedan prst samo kad mu Jovan to kaže. Kada Dule izbroji do pet on kaže Stanoju da izbroji još jedan prst na svojoj desnoj ruci, a Dule krene brojanje od početka.

Stanoje broji jedan po jedan prst samo kad mu Dule to kaže.

Kada završe brojanje svaki od njih prstima desne ruke pokazuje broj između 0 i 4. Napisati program koji pastirima pomaže da na osnovu broja otvorenih prstiju sva tri pastira redom izračunaju koliko stado ima ovaca.

Na standardnom ulazu u prvom redu su data 3 cela broja iz intervala [0,4] koja redom predstavljaju broj otvorenih prstiju desne ruke naših pastira Jovana, Duleta i Stanoja. Brojevi su razdvojeni blanko karakterom. Na standardnom izlazu u jednoj liniji prikazati broj ovaca u stadu.

Ulaz	Izlaz
3 1 3	83

**REŠENJE**

## I način

Svaki prst na Jovanovoj ruci predstavlja jednu ovcu. Svaki prst na Duletovoj ruci predstavlja 5 ovaca, zato što Dule otvara sledeći prst samo kad Jovan izbroji 5 ovaca. Svaki prst na Stanojevoj ruci predstavlja 25 ovaca, zato što Stanoje otvara sledeći prst samo kad Dule izboji do 5. Već smo objasnili da svaki Duletov prst predstavlja 5 ovaca, prema tome 5 Duletovih prstiju vredi  $5 * 5$  ovaca.

Obeležimo sa *rukaJ*, *rukaD*, *rukaS* broj otvorenih prstiju desne ruke redom kod Jovana, Duleta i Stanoja. Tada je broj ovaca u stadu jednak  $rukaS * 25 + rukaD * 5 + rukaJ$

```
#include <iostream>
using namespace std;
int main() {
    int rukaJ, rukaD, rukaS, brOvaca;
    cin >> rukaJ >> rukaD >> rukaS;
    brOvaca = rukaS * 25 + rukaD * 5 + rukaJ;
    cout << brOvaca;
    return 0;
}
```

Možemo primetiti da Jova i Dule ne mogu imati otvorena više od 4 prsta, jer svaki put kad izbroje do 5, počinju brojanje iz početka, a sledeći pastir otvara još jedan prst.

## II način

Ovo je primer brojanja u sistemu sa osnovom 5.

Jedan efikasan način da se izračuna vrednost ovog izraza je Hornerova šema koja polazi od vrednosti nula i u svakom koraku množi prethodnu vrednost osnovom 5 i na rezultat dodaje sledeću cifru.

```
#include <iostream>
using namespace std;
int main() {
    int rukaJ, rukaD, rukaS, brojOvaca;
    cin >> rukaJ >> rukaD >> rukaS;
    brojOvaca = 0;
    brojOvaca = brojOvaca*5 + rukaS;
    brojOvaca = brojOvaca*5 + rukaD;
    brojOvaca = brojOvaca*5 + rukaJ;
    cout << brojOvaca << endl;
    return 0;
}
```

10. Napisati program koji sa standardnog ulaza učitava brojilac i imenilac nepravog razlomka  $a/b$  i prevodi ga i mešoviti broj  $Nc/b$  tako da važi da je  $c/b < 1$  i ispisuje taj broj na standardni izlaz. U prvoj liniji standardnog ulaza dat je neprav razlomak u formi  $a/b$  tako da prirodan broj  $a$  predstavlja brojilac nepravog razlomka,  $a$  prirodan broj  $b$  predstavlja imenilac razlomka ( $a \geq b$ ). Prva i jedina linija standardnog izlaza sadrži mešoviti zapis razlomka, preciznije prirodan broj, brojilac i imenilac mešovitog broja međusobno odvojeni sa po jednom prazninom (blanko znakom).

Ulaz	Izlaz
23/8	2 7 8

## REŠENJE

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {  
    int a, b;  
    char crta;  
    cin >> a >> crta >> b;  
    int n = a / b;  
    int c = a % b;  
    cout << n << " " << c << " " << b << endl;  
    return 0;  
}
```