

Zadaci za samostalan rad

Grupa 2

Zadatak sa prethodnog časa - rešenje

1. <http://bee.bubblecup.org/JumpsJBOI2008>

Odabrali neki od 52 zadatka sa spiska na adresi

<http://mendo.mk/Training.do?cid=2>

Preporuka: Jumps, Map, Heritage

Prethodno se registrujte na MENDO sistemu.

Rekurentne relacije

Zeka mora da pređe n metara skačući i to skokovima dužine 3, 2 ili 1 metar. Na koliko načina to može da uradi zeka, ako dužine uspešnih skokova formiraju neopadajući niz? Napišite program koji računa broj načina. Vrednost n se unosi sa standardnog ulaza ($1 \leq n \leq 10^9$). Program treba da na standardni izlaz ispiše jedan ceo broj jednak ostatku pri deljenju sa 1000000.

Test primer

Input	Output
6	7

Objašnjenje: Broj različitih načina je 7, i njegov ostatak pri deljenju sa 1000000 je 7. Nizovi skokova su:

- 1) 3+3
- 2) 1+2+3
- 3) 1+1+1+3
- 4) 2+2+2
- 5) 1+1+2+2
- 6) 1+1+1+1+2
- 7) 1+1+1+1+1+1

Rešenje 1

Označimo sa x, y, z broj skokova dužine 3, 2, 1, redom.

Neka $M=1000000$

Mi zapravo treba da izračunamo broj različitih nenegativnih celobrojnih rešenja jednačine $3x + 2y + z = n$.

Jasno da $x \leq n/3$, i da za fiksiranu vrednost x , moguće vrednosti za y su $0, 1, \dots, (n - 3x)/2$.

```
int cnt=0;
for(int x=0; x<= n/3; x++)
for(int y=0; y<=(n-3*x)/2; y++)
{ cnt++;
if(cnt>=M) cnt = cnt - M;
}
```

```
cout << cnt << endl;
```

Mana ovog rešenja je kvadratna složenost.

Na primer, ako je $n = 987654321$, prevelika je složenost da bi dobili rešenje 324748.

Testirajmo program:

```
#include <iostream>
```

```

typedef long long ll;
const ll M = 1000000;
using namespace std;
int main() {
    int n;
    cin >> n;
    long long cnt=0; //resenje, broj nacina
    for(int x=0; x<= n/3; x++)
    for(int y=0; y<=(n-3*x)/2; y++)
    { cnt++;
    if(cnt>=M) cnt = cnt - M;
    }
    cout << cnt << endl;
    return 0;
}

```

Rešenje 2

Možemo ukloniti unutrašnji ciklus u rešenju 1. Za fiksiranu vrednost za x postoji tačno $1 + (n - 3x) / 2$ mogućnosti za y .

Dodatno da bi izbegli množenja, označimo da $a = 3x$. Novo rešenje:

```

int cnt=0;
for(int a=0; a<=n; a=a+3)
{ cnt = cnt + (n-a)/2 + 1;
if(cnt>M) cnt = cnt%M;
}
cout << cnt << endl;

```

Složenost $O(n)$

Rešenje 3

Koristeći prethodna dva načina za rešavanje ovog problema, možemo uočiti sledeće karakteristike konačnog rešenja:

Uočimo broj izračunatih načina za $n = 1, 7, 13, 19, \dots, 6k + 1, \dots$

n	1	7	13	19	25
cnt	1	$8 = 1 + 7$	$21 = 8 + 13$	$40 = 21 + 19$	$65 = 40 + 25$

Na primer za $n=7$, razbijanja broja 7 su

- 1) 3+3+1
- 2) 3+2+1+1
- 3) 3+1+1+1+1
- 4) 3+2+2
- 5) 2+2+2 +1
- 6) 2+2+1+1 +1
- 7) 2+1+1+1+1 +1
- 8) 1+1+1+1+1+1+1

Dakle, uočimo da rešenje za $x \geq 7$ zadovoljava rekurentnu relaciju $cnt(x) = cnt(x - 6) + x$.

Na primer, za $x=13$, $21=cnt(x)=cnt(7)+13=8+13$

Možemo pretpostaviti da $cnt(0)=1$

```

int x = n%6;
int cnt=0;

```

```

if(x==0) cnt=1; else cnt=x;
while(x<n)
{ x = x + 6;
cnt = cnt + x;
if(cnt >= M) cnt = cnt%M;
}
cout << cnt << endl;

```

Složenost $O(n)$, ali bez množenja u odnosu na rešenje 2

Rešenje 4

U odnosu na rešenje 3, uočimo da rešenje za $x=25$ je $cnt(25) = 1 + 7 + 13 + 19 + 25$,
Tj. $cnt(6k + 1) = 1 + 7 + 13 + \dots + (6k + 1)$.

Dakle, uvećanje svakog sabirka u sumi je 6, a broj sabiraka je $k + 1$.

Možemo primeniti Gausov trik da sračunamo svaku sumu. Na primer:

$$S = 1 + 7 + 13 + 19 + 25$$

$$S = 25 + 19 + 13 + 7 + 1$$

$$2S = 26 + 26 + 26 + 26 + 26 = 26 * 5$$

$$S = 26 * 5 / 2 = 65.$$

$$\text{Za } n=6k+1 \text{ imamo da } S=(1+n)/2*(1+k)$$

Implementirajmo ovu ideju:

```

long long cnt = ((n%6 + n)/2%M * ((1 + n/6)%M));
if(n%6==0) cnt++;
cout << cnt%M << endl;

```

Uocite da smo napravili malu korekciju resenja za slucaj kad je n deljivo sa 6, tj.

```

if(n%6==0) cnt++;

```

Još test primera

ULAZ	IZLAZ
15	27
234	4681
123456789	244272
987654321	324748
111222333	441408
100500700	291184
88888888	267490

```

#include <iostream>
typedef long long ll;
const ll M = 1000000;
using namespace std;
int main() {
int n;
cin >> n;
long long cnt = ((n%6 + n)/2%M * ((1 + n/6)%M));
if(n%6==0) cnt++;
cout << cnt%M << endl;
return 0;
}

```

Mentorska nastava – zadaci za samostalni rad

2.

<http://bee.bubblecup.org/PRAVOUGAONIKSIO2014>

Neka P je pravougaonik čije sve stranice imaju celobrojne dužine. Pravougaonik je izdeljen na jedinične kvadratiće (1×1). Označimo sa $f(P)$ broj jediničnih kvadratića koji imaju zajedničku tačku sa jednom dijagonalom pravougaonika. Na primer, ako stranice pravougaonika P su dužine 4 i 2, onda $f(P) = 4$.

Napišite program PRAVOUGAONIK koji će za dati pozitivan ceo broj N izračunati broj različitih pravougaonika P za koje važi $f(P) = N$. Smatrati da dva pravougaonika čije su stranice $a \times b$ i $b \times a$ nisu različiti.

Ulaz

U jedinoj liniji standardnog ulaza zadat je ceo broj N ($0 < N < 1000000$).

Izlaz

U jedinu liniju standardnog izlaza upisati jedan ceo broj – izračunati broj pravougaonika.

PRIMER

Ulaz Izlaz

4 4

Objašnjenje: Broj različitih pravougaonika P za koje važi da $f(P) = 4$ jeste 4. Ta četiri pravougaonika imaju redom stranice sledeće dužine: 1×4 , 2×3 , 2×4 , 4×4 .

Vremensko ograničenje 1s

Memorijsko ograničenje 64 MB

3. <http://bee.bubblecup.org/TelefonUlica#problem>

U naselju ima n zgrada, raspoređenih po pravom putu s jedne strane na jednakim (jediničnim) rastojanjima. U naselje se uvodi telefonska veza. U nizu T od n elemenata, navedeno je koliko priključaka treba uvesti u svaku zgradu. Svaki telefon mora biti vezan sa centralom posebnim provodnikom. Centrala će biti postavljena u neku od postojećih zgrada. Odrediti kolika je minimalna zbirna dužina provodnika od centrale do svih priključaka.

Input

U prvom redu broj zgrada n , manji ili jednak milion. U drugom redu n celih nenegativnih brojeva $T[i]$, manjih ili jednakih 1000, razdvojenih razmakom.

Output

Jedan broj, tražena minimalna zbirna dužina provodnika.

Example

Example Input	Example Output
6	30

3 5 1 6 2 4	
-------------	--

Example Description

Minimalna zbirna dužina provodnika se dobija kada se centrala smesti u četvrtu zgradu sleva, a tada je potrebno $3 \cdot 3 + 2 \cdot 5 + 1 \cdot 1 + 1 \cdot 2 + 2 \cdot 4 = 30$ dužinskih jedinica provodnika.

4. <http://bee.bubblecup.org/Problems/BC3NSUB>

Dat je niz A dužine N. Za podniz $(a[i], a[i+1], \dots, a[j]) = a[i, j]$, gde je $i < j$, kažemo da je lep ukoliko važi:

- $a[i] \leq a[j]$
- $a[i] \leq a[k] \leq a[j]$, za svako k iz $[i, j]$

Napisati program koji nalazi najduži lepi podniz za dati niz prirodnih brojeva.

Input

U prvom redu ulaza se nalazi prirodni broj N koji predstavlja dužinu niza. U narednih N redova se nalazi po jedan prirodan broj - elementi datog niza.

Output

U prvom i jedinom redu štampati dužinu najdužeg lepog podniza.

Example

Example Input	Example Output
6 1 3 4 2 5 0	5

Constraints

- $2 \leq N \leq 10^6$
- $0 \leq a[i] \leq 2 \cdot 10^9$, za i iz $[1, n]$

Grupa 1 – zadaci sa rešenjima sa prethodnog časa

1. Komutativnost

Greške u programima koje se tiču računskih izraza je nekad tesko uočiti. Jedna od očiglednih stvari na koje treba obratiti pažnju je redosled operanada u izrazu - nekad nije važan, nekad jeste. Dakle, koje od navedenih računskih operacija u C-u su komutativne?

- a. +
- b. -
- c. *
- d. /

Rezultat prikaži kao nisku karaktera u intervalu [a,d] u jednom redu, bez razmaka, navodeći samo slova uz komutativne operacije. Npr. ako su sve operacije komutativne, ispisati:

abcd

IZLAZ ac

2. Poredjaj vrednosti izraza

Izrazi u programskom jeziku nisu uvek isto što i matematički izrazi... Uporedi vrednosti navedenih izraza

- a. $1 + 1/2$
- b. $4*(3/8)$
- c. $2*(1.0/4+2/4)$
- d. 1.5

Rezultat prikazati kao nisku karaktera u intervalu [a,d] u jednom redu, bez razmaka, navodeći slova uz vrednosti u rastucem poretku. Npr. Ako su izrazi u opadajućem poretku, ispisati:

dcba

IZLAZ bcad

3. Benzin ili dizel

Komšija zeli da kupi nov auto, ali ima problem da se opredeli da li da kupi auto koji ide na benzin ili dizel, pa te je zamolio da mu pomogneš da izračuna šta mu se više isplati. Dizel auto je skuplji za 125000 dinara od ekvivalentnog benzinskog auta. Dizel auto troši 4 litra na 100 kilometara otvorenog puta, a 8 l na 100km u gradu. Benzinski auto troši 5 l na 100km i 10 l na 100km, na otvorenom i po gradu, redom. Komsija planira da vozi auto 5 godina, godisnje predje 10000km po gradu i jos 12000km na otvorenom putu. Cena 1l benzina je 131.40 dinara, a dizela 143.00. Konacno, servis dizel vozila je 10000 dinara godisnje skuplji od servisa benzinskog vozila, ali je registracija 5000 din jeftinija. Pretpostavka je da su cene goriva i registracije konstantne.

Izlaz: ispisati razliku u iznosu, u dinarima, koju bi u roku od 5 godina komsija platio za benzinsko vozilo u odnosu na dizel. (Izlaz je negativna vrednost ako je benzinski auto jeftiniji, pozitivan ako je dizel jeftiniji)

```
#include <iostream>
```

```
#include <cmath>
```

```

using namespace std;
int main()
{
    //rezultat je benzinacPotrosnja – (125000+dizelPotrosnja +10000-5000)
    /* benzinacPotrosnja=5 godina * (gradska voznja*131.4 +otvorenPutVoznja*131.4)=
    5*(10*10000/100*131.4 + 5*12000/100*131.4) */

    cout << 5*(10*100*131.4+5*120*131.4)-
        125000- 5*(8*100*143+4*120*143+5000);
    return 0;
}

```

IZLAZ -14000

4. Merdevine

Mali Đokica želi da opere prozore na prvom spratu svoje kuće, koji su na visini od 4 metara. Strana kuće na kojoj su prozori ima baštu sa cvećem neposredno ispred zida. Bašta se proteže celom dužinom i široka je 1.5 metara. U baštu je uloženo mnogo truda, i Đokici nije dozvoljeno da merdevine postavi među cvećem. Đokici treba pomoć da odredi koja je minimalna dužina merdevina koje će da kupi tako da može da opere prozor. Na izlazu ispisati dužinu merdevina u metrima.

IZLAZ 4.272

Pitagorina teorema!!!

C++ rešenje	C rešenje
<pre> #include <iostream> #include <cmath> using namespace std; int main() { cout << sqrt(4*4+1.5*1.5); return 0; } </pre>	<pre> #include <stdio.h> #include <math.h> int main() { printf("%f", sqrt(4*4+1.5*1.5)); return 0; } </pre>

Grupa 1 – zadaci za samostalni rad

5. <http://bubblebee.rs/Problems/DefinicijaPromenljive>

Ispraviti program koji čita i ispisuje jedan ceo broj.

```

int main()
{
    cin >> x;
    cout << x;
}

```

6. <http://bubblebee.rs/Problems/NedozvoljenoIme>

Ispraviti zadatak koji računa zbir brojeva od 1 do 6.

```
int main()
{
    int ia = 1;
    int ib = 2;
    int ic = 3;
    int id = 4;
    int ie = 5;
    int if = 6;
    cout << ia+ib+ic+id+ie+if;
}
```

7. <http://bubblebee.rs/Problems/PoslednjiSkolskiCas>

Danas ti prvi školski čas počinje u 08:00 AM. Jedan školski čas traje 45 minuta, mali odmor između časova traje 5 minuta a veliki odmor traje 20 minuta. Veliki odmor je nakon 2. časa a mali odmori su nakon svih ostalih.

Ako danas imaš N časova (a sigurno ćeš ih imati više od 2 ☺), koliko će biti sati i koliko će biti minuta nakon što ti se završi poslednji čas?

Input

N (broj današnjih časova)

Output

SS:MM (sati:minuti)

Example

Example Input	Example Output
6	13:10

8. <http://bubblebee.rs/Problems/UnosIspis>

Pročitati dva broja i ispisati te iste brojeve.

Input

Dva cela broja.

Output

Dva pročitana cela broja.

Example

Example Input	Example Output
5 11	5 11

9. <http://bubblebee.rs/Problems/KonverzijaVremena>

Pretvoriti vreme iz sekundi u sate, minute i sekunde.

Input

Vreme u sekundama.

Output

Pretovreno vreme u satima, minutima i sekundama.

Example

Example Input	Example Output
20051	5:34:11

NAPOMENA: Uvidom u test primere, konstatujem da autor test primera nije vodio racuna o vodecim nulama. tj. nema primera sa jednocifrenim vrednostima minuta i sekundi!!!

Osmislite sami nekoliko takvih primera!

```
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    int t;
    cin >> t;
    int h=t/3600;
    t=t%3600;
    int m=t/60;
    int s=t%60;
    /*cout.width(2);
    cout.fill('0');
```

```
*/  
cout << h << ":";  
  
cout.width(2);  
cout.fill('0');  
  
cout << m << ":";  
  
cout.width(2);  
cout.fill('0');  
  
cout << s;  
return 0;  
}
```

10. <http://bee.bubblecup.org/Problems/Pravougaonik>

Izračunati obim i površinu pravougaonika ako su date dužine stranica.

Input

Dužine stranica pravougaonika.

Output

Obim i površina pravougaonika.

Example

Example Input	Example Output
3 9	24 27

11.

<http://bee.bubblecup.org/Problems/ProtekloVreme>

Izračunati vreme proteklo između dva trenutka.

Input

HH:MM:SS HH:MM:SS

Output

HH:MM:SS

Example

Example Input	Example Output
06:36:46 08:38:48	02:02:02

Note

Kada se učitava podatak tipa **int**, preskaču se svi nevidljivi znakovi sa ulaza (tipa razmak), i učitace se svi znakovi do prvog znaka koji nije cifra. Da se uči samo jedan znak sa ulaza treba da se uči podatak tipa **char**. Primera radi, za ulaz "1m65cm", sledeći kod:

```
void main()
{
    int metri;
    int centimetri;
    char znak;
    cin >> metri >> znak >> centimetri;
    cout << metri << " metara " << centimetri << " centimetara."
}
ispisuje "5 metara 65 centimetara."
```

12.

<http://bee.bubblecup.org/Problems/Separator#istorija>

Ispraviti program koji čita i ispisuje jedan ceo broj.

```
int main()
{
    int x;
    cin >> x
    cout << x;
}
```