

Uzdevumu portfolio

Portfolio iekļauti 50 paaugstinātas grūtības pakāpes uzdevumi no clevercode.lv un olimps.lv saitēm.

<https://github.com/kevnix/skolasuzd>

Burtu klucīši

Apraksts

Rotai ir klucīši ar burtu attēliem. Uz katra klucīša ir uzzīmēts viens latīņu alfabēta lielais burts. Uz vairākiem klucīšiem var būt uzzīmēts viens un tas pats burts. Rota no saviem klucīšiem, saliekot tos rindā vienu aiz otra, ir izveidojusi kādu vārdu. Rotas draugs Zigmārs arī vēlētos salikt kādu vārdu, diemžēl viņam nav šādu klucīšu. Rota ir ar mieru aizdot tikai tos klucīšus, no kuriem ir salikts viņas iedomātais vārds, un nevienu vairāk.

Uzrakstiet programmu, kas dotiem Rotas saliktā un Zigmāra iedomātajiem vārdiem nosaka, vai Zigmārs no dotajiem klucīšiem varēs salikt savu iedomāto vārdu.

Risinājums

```
#include <iostream>
#include <string>
#include <algorithm>

using namespace std;

int main() {
    string rota, vards;
    cin >> rota >> vards;
    sort(rota.begin(), rota.end()); // sakārto Rotas klucīšus alfabētiski
    sort(vards.begin(), vards.end()); // sakārto Zigmāra vārda burtus alfabētiski

    bool varSalikt = true;
    int i = 0, j = 0;
    while (i < rota.length() && j < vards.length()) {
        if (rota[i] == vards[j]) {
            i++;
            j++;
        } else if (rota[i] < vards[j]) {
            i++;
        } else {
            varSalikt = false;
            break;
        }
    }
    if (j < vards.length()) {
        varSalikt = false;
    }
    if (varSalikt) {
        cout << "VAR" << endl;
    } else {
        cout << "NEVAR" << endl;
    }
    return 0;
}
```

Rezultāts (12/12 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
545457150f2b7	Pabeigts	0.00	2.0 MB	Jā
545457150f32c	Pabeigts	0.00	2.0 MB	Jā
545457150f3af	Pabeigts	0.00	2.0 MB	Jā
545457150f41b	Pabeigts	0.00	2.0 MB	Jā
545457150f486	Pabeigts	0.00	2.0 MB	Jā
545457150f4f7	Pabeigts	0.00	2.0 MB	Jā
545457150f567	Pabeigts	0.00	2.0 MB	Jā
545457150f5d6	Pabeigts	0.00	2.0 MB	Jā
545457150f647	Pabeigts	0.00	2.0 MB	Jā
545457150f6ba	Pabeigts	0.00	2.0 MB	Jā

A burtu skaits vārdā

Apraksts

Uzdevums sameklēt burtu 'a' skaitu dotajā vārdā.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    string vards;
    int x=0;
    cin>>vards;
    for(int i=0; i<vards.length(); i++){
        if(vards[i]=='a'){
            x++;
        }
    }
    cout<<x;
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5627f0bb46aeb	Pabeigts	0.00	2.0 MB	Jā
5627f0e6c7b17	Pabeigts	0.00	2.0 MB	Jā
5627f0f9246ac	Pabeigts	0.00	2.0 MB	Jā
5627f11a720a1	Pabeigts	0.00	2.0 MB	Jā
5627f125e80be	Pabeigts	0.00	2.0 MB	Jā

Vārds no burtiem

Apraksts

Izdrukāt vārdu, kas veidojas no citu vārdu pirmajiem burtiem.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    string vards, salikts="";

    for(int i=0; i<5; i++){
        cin>>vards;
        salikts +=vards[0];
    }
    cout<<salikts;
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5658cda213354	Pabeigts	0.00	2.0 MB	Jā
5658ce091b475	Pabeigts	0.00	2.0 MB	Jā
5658ce8d49633	Pabeigts	0.00	2.0 MB	Jā
5658ced297880	Pabeigts	0.00	2.0 MB	Jā
5658cf2a99ec1	Pabeigts	0.00	2.0 MB	Jā

Pēdējais burts vārdā

Apraksts

Atrast vārda pēdējo burtu.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    string vards;
    cin>>vards;
    cout<<vards[vards.length()-1];
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5658d3a980cae	Pabeigts	0.00	2.0 MB	Jā
5658d3c58d39e	Pabeigts	0.00	2.0 MB	Jā
5658d3d1e07b4	Pabeigts	0.00	2.0 MB	Jā
5658d3e6047fb	Pabeigts	0.00	2.0 MB	Jā
5658d3fce7022	Pabeigts	0.00	2.0 MB	Jā

Dotā burta skaits vārdā

Apraksts

Atrast cik reižu dotais burts atkārtojas dotajā vārdā.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    string vards;
    char burts;
    int x=0;
    cin>>vards;
    cin>>burts;

    for(int i=0; i<vards.length(); i++){
        if(vards[i]==burts) x++;
    }
    cout<<x;
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5659803b7eff8	Pabeigts	0.00	2.0 MB	Jā
56598074ea915	Pabeigts	0.00	2.0 MB	Jā
5659808e932fb	Pabeigts	0.00	2.0 MB	Jā
5659809a3ba0a	Pabeigts	0.00	2.0 MB	Jā
5659805410f77	Pabeigts	0.00	2.0 MB	Jā

Kurš burts pirmais?

Apraksts

Dots vārds, kura garums nepārsniedz 20 burtus un 2 burti. Atrast un izvadīt to burtu no diviem dotajiem, kurš, lasot doto vārdu, parādās pirmais. Zināms, ka katrs no burtiem vārdā sastopams tieši vienu reizi.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    string vards;
    char b1, b2;

    cin>>vards;
    cin>>b1;
    cin>>b2;

    for(int i=0; i<vards.length(); i++){
        if(vards[i]==b1){
            cout<<b1;
            return 0;
        }else if(vards[i]==b2){
            cout<<b2;
            return 0;
        }
    }
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
566b4d72316ac	Pabeigts	0.00	2.0 MB	Jā
566b4d90d85f6	Pabeigts	0.00	2.0 MB	Jā
566b4dfb92e65	Pabeigts	0.00	2.0 MB	Jā
566b4e86dd7f1	Pabeigts	0.00	2.0 MB	Jā
566b4eefaebc2	Pabeigts	0.00	2.0 MB	Jā

Vārda veidošana

Apraksts

Doti 2 vārdi, kuru garums nepārsniedz 20 burtus. Pārbaudīt vai no pirmā vārda burtiem var izveidot otro vārdu. Zināms, ka otrais vārds ir īsāks par pirmo un tajā neviens burts neatkārtojas. Atbildes ir Var un Nevar.

Risinājums

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    string v1, v2;
    char b;
    bool var = true;
    cin >> v1 >> v2;

    for (int i=0; i<v2.length(); i++) {
        b=v2[i];
        if (v1.find(b) == string::npos) {
            var = false;
            break;
        }
    }
    if (var) {
        cout << "Var" << std::endl;
    } else {
        cout << "Nevar" << std::endl;
    }
    return 0;
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
566b567fb6904	Pabeigts	0.00	2.0 MB	Jā
566b585975ee0	Pabeigts	0.00	2.0 MB	Jā
566b56c345a8d	Pabeigts	0.00	2.0 MB	Jā
566b574274395	Pabeigts	0.00	2.0 MB	Jā
566b57bd829af	Pabeigts	0.00	2.0 MB	Jā

Aizstāj burtu

Apraksts

Dots vārds un 2 burti. Uzdevums sameklēt dotajā vārdā visus burtus, kas vienādi ar pirmo doto burtu un aizstāt tos ar otro doto burtu.

Risinājums

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    string vards;
    char b1, b2;
    cin>>vards>>b1>>b2;
    for(int i=0; i<vards.length(); i++){
        if(vards[i]==b1){
            vards[i]=b2;
        }
    }
    cout<<vards;
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
56921077ceef8	Pabeigts	0.00	2.0 MB	Jā
5692109c5d7eb	Pabeigts	0.00	2.0 MB	Jā
569210c54b2e5	Pabeigts	0.00	2.0 MB	Jā
569210fed9a6b	Pabeigts	0.00	2.0 MB	Jā
56921127e6ef6	Pabeigts	0.00	2.0 MB	Jā

Vārds no pirmajiem burtiem

Apraksts

Dots noteikts vārdu skaits. Izveidot vārdu no doto vārdu pirmajiem burtiem.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    string vards, salikts;
    int sk;
    cin>>sk;
    for(int i=0; i<sk; i++){
        cin>>vards;
        salikts+=vards[0];
    }
    cout<<salikts;
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
59afa20a49fa8	Pabeigts	0.00	2.0 MB	Jā
59afa21b7cb07	Pabeigts	0.00	2.0 MB	Jā
59afa24217951	Pabeigts	0.00	2.0 MB	Jā
59afa252451f3	Pabeigts	0.00	2.0 MB	Jā
59afa2641a67c	Pabeigts	0.00	2.0 MB	Jā

Sakārto vārda burtus

Apraksts

Dots vārds. Izveido jaunu vārdu, kas veidots no dotā vārda burtiem alfabētiskā secībā.

Risinājums

```
#include <iostream>
#include <algorithm>
using namespace std;
int main() {
    string vards;
    cin>>vards;
    sort(vards.begin(), vards.end());
    cout<<vards;
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
59c0c981e1968	Pabeigts	0.00	2.0 MB	Jā
59c0c9b8d58a2	Pabeigts	0.00	2.0 MB	Jā
59c0c9cc4488e	Pabeigts	0.00	2.0 MB	Jā
59c0ca1178f5f	Pabeigts	0.00	2.0 MB	Jā
59c0c9eed102c	Pabeigts	0.00	2.0 MB	Jā

Cikls1 iesācējiem

Apraksts

Jāizdrukā vārds CIKLS noteiktu reižu skaitu!

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int a;
    cin>>a;

    for(int i=0; i<a; i++){
        cout<<"CIKLS"<<endl;
    }
}
```

Rezultāts (3/3 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
560ece23d49dd	Pabeigts	0.00	2.0 MB	Jā
560ece23d4a90	Pabeigts	0.00	2.0 MB	Jā
560ece23d4b37	Pabeigts	0.00	2.0 MB	Jā

Cikls2 iesācējiem

Apraksts

Ielasi skaitli un vārdu. Izvadi nolasīto vārdu tik reizes, cik liels ir nolasītais skaitlis.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int a;
    string v;
    cin>>a;
    cin>>v;

    for(int i=0; i<a; i++){
        cout<<v<<endl;
    }
}
```

Rezultāts (4/4 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
560ed0a4e599a	Pabeigts	0.00	2.0 MB	Jā
560ed0a4e5a52	Pabeigts	0.00	2.0 MB	Jā
560ed0a4e5b12	Pabeigts	0.00	2.0 MB	Jā
560ed0e86caf2	Pabeigts	0.00	2.0 MB	Jā

Cikls3 iesācējiem

Apraksts

Atrodi cik reižu meklējamais skaitlis sakrīt ar kādu no dotajiem skaitļiem!

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int a, b, c, x=0;
    cin>>a>>b;

    for(int i =0; i<a; i++){
        cin>>c;
        if(c==b) x++;
    }
    cout<<x;
}
```

Rezultāts (3/3 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
560ed29d1c049	Pabeigts	0.00	2.0 MB	Jā
560ed29d1c112	Pabeigts	0.00	2.0 MB	Jā
560ed29d1c1b4	Pabeigts	0.00	2.0 MB	Jā

Cikls4 iesācējiem

Apraksts

Izdrukāt visus skaitļus no 1 līdz dotajam skaitlim.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int a;
    cin>>a;

    for(int i=1; i<=a; i++){
        cout<<i<<endl;
    }
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
56323da070f6d	Pabeigts	0.00	2.0 MB	Jā
56323dbbd2efd	Pabeigts	0.00	2.0 MB	Jā
56323dc205125	Pabeigts	0.00	2.0 MB	Jā
56323dd612577	Pabeigts	0.00	2.0 MB	Jā
56323dd612b7a	Pabeigts	0.00	2.0 MB	Jā

Cikls5 iesācējiem

Apraksts

Jāatrod visu pēc kārtas ņemtu veselo skaitļu summa no viena skaitļa līdz otram.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int a, b, sum=0;
    cin>>a>>b;

    for(int i=a; i<=b; i++){
        sum+=a;
        a++;
    }
    cout<<sum;
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5632adb4dfe0a	Pabeigts	0.00	2.0 MB	Jā
5632adcc29397	Pabeigts	0.00	2.0 MB	Jā
5632ade7edd17	Pabeigts	0.00	2.0 MB	Jā
5632ae299745f	Pabeigts	0.00	2.0 MB	Jā
5632ae3c35db7	Pabeigts	0.00	2.0 MB	Jā

Cikls6 iesācējiem

Apraksts

Doti divi skaitļi. Izdrukāt mazāko skaitli tik reižu, cik liels ir lielākais skaitlis.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int a, b;
    cin>>a>>b;

    if(a>b){
        for(int i=0; i<a; i++){
            cout<<b<<endl;
        }
    }else{
        for(int i=0; i<b; i++){
            cout<<a<<endl;
        }
    }
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
59af9d3ec151c	Pabeigts	0.00	2.0 MB	Jā
59af9d9e09f1b	Pabeigts	0.00	2.0 MB	Jā
59af9dd7b3cb6	Pabeigts	0.00	2.0 MB	Jā
59af9df7262d9	Pabeigts	0.00	2.0 MB	Jā
59af9e1837108	Pabeigts	0.00	2.0 MB	Jā

Ciparu summa

Apraksts

Atrast vesela skaitļa ciparu summu!

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    long a, cipsum=0;
    cin>>a;
    while(a>0){
        cipsum+=a%10;
        a/=10;
    }
    cout<<cipsum;
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
571dc69a2188e	Pabeigts	0.00	2.0 MB	Jā
571dc6a6666b5	Pabeigts	0.00	2.0 MB	Jā
571dc6cdc7284	Pabeigts	0.00	2.0 MB	Jā
571dc8c9b465b	Pabeigts	0.00	2.0 MB	Jā
571dc8eb0c844	Pabeigts	0.00	2.0 MB	Jā

$$2 + 2$$

Apraksts

Viena no visvieglākajām problēmām starp sporta programmēšanas problēmām ir divu skaitļu saskaitīšana. Ievaddatos tiek doti divi skaitļi, uzraksti programmu, kas saskaita tos.

Risinājums

```
#include <iostream>
int main() {
    int n, m;
    std::cin >> n >> m;
    std::cout<<n+m;
}
```

Rezultāts (6/6 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5435b29f8c2c4	Pabeigts	0.00	2.0 MB	Jā
5435b29f8c564	Pabeigts	0.00	2.0 MB	Jā
5435b4aeda23a	Pabeigts	0.00	2.0 MB	Jā
5435b4aeda39e	Pabeigts	0.00	2.0 MB	Jā
54415a0a7e67f	Pabeigts	0.00	2.0 MB	Jā
54415a0a7e7c4	Pabeigts	0.00	2.0 MB	Jā

Dīvainā summa

Apraksts

Doti n veseli nenegatīvi skaitļi. Jāatrod šo skaitļu dīvainā summa, izmantojot tālāk doto algoritmu.

Ja skaitlis ir nepāra skaitlis, tad tas tiek vienkārši pieskaitīts summai.

Ja skaitlis ir pāra skaitlis, tad tas tiek izdalīts ar 2 un pieskaitīts summai.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int n, a, sum=0;
    do{
        cin>>n;
    }while(n<1||n>100);

    for(int i=0; i<n; i++){
        do{
            cin>>a;
        }while(a<1||a>100);

        if(a%2==0) sum+=a/2;
        else sum+=a;
    }
    cout<<sum;
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5659541150768	Pabeigts	0.00	2.0 MB	Jā
565954580ee71	Pabeigts	0.00	2.0 MB	Jā
56595489158a2	Pabeigts	0.00	2.0 MB	Jā
565953e99fa21	Pabeigts	0.00	2.0 MB	Jā
56595430eca58	Pabeigts	0.00	2.0 MB	Jā

Garākais vārds

Apraksts

Atrast garāko no 2 dotajiem vārdiem. Ja garāks ir pirmais vārds vai vārdi ir vienāda garuma, tad atbilde ir „Pirmais”, bet pretējā gadījumā atbilde ir „Otrais”.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    string v1, v2;
    do{
        cin>>v1;
        cin>>v2;
    }while(v1.length()>20 || v2.length()>20);

    if(v1.length()>v2.length()){
        cout<<"Pirmais";
    }else if(v1.length()<v2.length()){
        cout<<"Otrais";
    }else{
        cout<<"Pirmais";
    }
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
566b46ec99792	Pabeigts	0.00	2.0 MB	Jā
566b470b62d65	Pabeigts	0.00	2.0 MB	Jā
566b473643572	Pabeigts	0.00	2.0 MB	Jā
566b474cf13be	Pabeigts	0.00	2.0 MB	Jā
566b4766c2052	Pabeigts	0.00	2.0 MB	Jā

Masīva kārtošana

Apraksts

Doti n ($1 < n \leq 10$) veseli skaitļi robežās no -100 līdz 100. Uzdevums sakārtot tos nedilstošā secībā un izvadīt vienu zem otra.

Risinājums

```
#include <iostream>
#include <algorithm>
using namespace std;
int main() {
    int n;
    cin>>n;
    int array[n];

    for(int i=0; i<n; i++){
        cin>>array[i];
    }
    sort(array, array + n);
    for(int i=0; i<n; i++){
        cout<<array[i]<<endl;
    }
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
566bf1893839e	Pabeigts	0.00	2.0 MB	Jā
566bf1d85e5b5	Pabeigts	0.00	2.0 MB	Jā
566bf209a7626	Pabeigts	0.00	2.0 MB	Jā
566bf249d546c	Pabeigts	0.00	2.0 MB	Jā
566bf25e823a2	Pabeigts	0.00	2.0 MB	Jā

Sakārto vārdus alfabēta secībā

Apraksts

Doti n ($1 < n < 10$) skolēnu vārdi. Sakārtot tos alfabēta secībā.

Risinājums

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

int main() {
    int n;
    cin >> n;
    string vardi[n];

    for(int i=0; i<n; i++){
        cin >> vardi[i];
    }

    sort(vardi, vardi+n);

    for(int i=0; i<n; i++){
        cout << vardi[i] << endl;
    }
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
566bf7b6ce390	Pabeigts	0.00	2.0 MB	Jā
566bfbe155083	Pabeigts	0.00	2.0 MB	Jā
566bfbf83b4c9	Pabeigts	0.00	2.0 MB	Jā
566bfc20a406d	Pabeigts	0.00	2.0 MB	Jā
566bfc65d783d	Pabeigts	0.00	2.0 MB	Jā

Nulles ietekme

Apraksts

Doti veseli skaitļi. Ja vismaz viens no dotajiem skaitļiem ir 0, tad jānomaina visu pārējo skaitļu zīme uz pretējo un jāizvada skaitļi augošā secībā. Pretējā gadījumā vienkārši jāizvada skaitļi augošā secībā.

Risinājums

```
#include <iostream>
#include <algorithm>
using namespace std;
int main() {
    int n;
    cin>>n;
    int array[n];
    bool nulle=false;

    for(int i=0; i<n; i++){
        cin>>array[i];
        if(array[i]==0){
            nulle=true;
        }
    }

    if(nulle==true){
        for(int i=0; i<n; i++){
            if(array[i]!=0){
                array[i] = -array[i];
            }
        }
    }

    sort(array, array+n);
    for(int i=0; i<n; i++){
        cout<<array[i]<<endl;
    }

}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
59c0cf44e430d	Pabeigts	0.00	2.0 MB	Jā
59c0cf80c1167	Pabeigts	0.00	2.0 MB	Jā
59c0cfcdb2139	Pabeigts	0.00	2.0 MB	Jā
59c0cfe7bb0e5	Pabeigts	0.00	2.0 MB	Jā
59c0d016e1aa9	Pabeigts	0.00	2.0 MB	Jā

Vārdu maiņa

Apraksts

Doti divi vārdi un n ($1 < n < 100$) vārdu saraksts. Visi saraksta vārdi, kas sakrīt ar pirmo vārdu, jāaizstāj ar otro vārdu. Pārveidotais saraksts jāizdrukā alfabētiskā secībā, drukājot vienu vārdu zem otra.

Risinājums

```
#include <iostream>
#include <algorithm>
using namespace std;
int main() {
    string v1, v2;
    int n;
    cin>>v1>>v2;
    cin>>n;
    string list[n];

    for(int i=0; i<n; i++){
        cin>>list[i];
        if(list[i]==v1){
            list[i]=v2;
        }
    }
    sort(list, list+n);

    for(int i=0; i<n; i++){
        cout<<list[i]<<endl;
    }
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
59ca99fde3463	Pabeigts	0.00	2.0 MB	Jā
59ca9a3f4fe92	Pabeigts	0.00	2.0 MB	Jā
59ca9adba0fa2	Pabeigts	0.00	2.0 MB	Jā
59ca9b1b01c43	Pabeigts	0.00	2.0 MB	Jā
59ca9c0569f3a	Pabeigts	0.00	2.0 MB	Jā

Skaitļu kārtošana

Apraksts

Jūsu uzdevums ir sakārtot dotus veselus skaitļus augošā secībā.

Risinājums

```
#include <iostream>
#include <algorithm>
using namespace std;
int main() {
    int n;
    cin>>n;
    int array[n];

    for(int i=0; i<n; i++){
        cin>>array[i];
    }
    sort(array, array + n);
    for(int i=0; i<n; i++){
        cout<<array[i]<<endl;
    }
}
```

Rezultāts (10/11 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
54545787c8fc6	Pabeigts	0.00	2.0 MB	Jā
54545787c906e	Pabeigts	-	2.2 MB	Nē
54545787cdb4c	Pabeigts	0.00	2.0 MB	Jā
54545787cdbef	Pabeigts	0.00	2.0 MB	Jā
54545787cdcb9	Pabeigts	0.00	2.0 MB	Jā
54545787cdd9b	Pabeigts	0.00	2.0 MB	Jā
54545787cdea2	Pabeigts	0.00	2.0 MB	Jā
54545787ce030	Pabeigts	0.00	2.0 MB	Jā
54545787ce525	Pabeigts	0.00	2.0 MB	Jā
54545787cee0a	Pabeigts	0.00	2.0 MB	Jā

Trokšņojošie lācīši

Apraksts

Katrā avīzes Postimees izdevumā tiek drukāta komiksa "Trokšņojošie lācīši" kāda daļa. Diemžēl dažreiz redakcijai gadās kļūdas un viena un tā pati bilde tiek drukāta vairākas reizes. Jūs, būdams "Lācīšu" piekritējs, vēlaties uzzināt, cik jaunu (tādu, kas nav izdoti agrāk) komiksu jūs esat dabūjis pagājušajā gadā. Par laimi, jums ir saglabājušās WWW-saites visa gada komiksiem. Visas saites ir sekojoša veida:

<http://www.postimees.ee/koomiks/karudXXXX.jpg>

XXXX ir komiksa zīmēšanas (nevis izdošanas) datums. Tā kā ir zināms, ka komiksu autors Urmass Nemvaltss nekad nezīmē divas bildes vienā dienā, tad pietiek uzrakstīt programmu, kas izrēķina dažādo datumu skaitu šajā sarakstā.

Risinājums

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

int main() {
    int N;
    cin >> N;
    set<string> komix;
    for (int i = 0; i < N; i++) {
        string url;
        cin >> url;
        //string date_str = url.substr(37,4);
        //cout<<date_str << " ";
        komix.insert(url);
    }
    cout << komix.size() << endl;
    return 0;
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
545457d08cddc	Pabeigts	0.00	2.0 MB	Jā
545457d08ce5f	Pabeigts	0.00	2.0 MB	Jā
545457d08ced1	Pabeigts	0.00	2.0 MB	Jā
545457d08cf72	Pabeigts	0.00	2.0 MB	Jā
546689b78d774	Pabeigts	0.00	2.0 MB	Jā

Lielāki par vidējo

Apraksts

Ievadītiem N naturāliem skaitļiem noteikt cik no šiem skaitļiem ir lielāki par visu ievadīto skaitļu aritmētisko vidējo. Piemēram, ja ievadīti skaitļi 3, 5 un 4, tad tikai viens no šiem skaitļiem (5) ir lielāks par visu skaitļu aritmētisko vidējo (4).

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int n, sum=0, vid=0, sk=0;
    cin>>n;

    int a[n];
    for(int i=0; i<n; i++){
        cin>>a[i];
        sum+=a[i];
    }
    vid=sum/n;

    for(int i=0; i<n; i++){
        if(a[i]>vid) sk++;
    }
    cout<<sk;

}
```

Rezultāts (10/10 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5454573f0a9d8	Pabeigts	0.00	2.0 MB	Jā
5454573f0aa60	Pabeigts	0.00	2.0 MB	Jā
5454573f0aace	Pabeigts	0.00	2.0 MB	Jā
5454573f0ab5b	Pabeigts	0.00	2.0 MB	Jā
5454573f0abf1	Pabeigts	0.00	2.0 MB	Jā
5454573f0ac61	Pabeigts	0.00	2.0 MB	Jā
5454573f0acd2	Pabeigts	0.00	2.0 MB	Jā
5454573f0ad42	Pabeigts	0.00	2.0 MB	Jā
5467122c901cc	Pabeigts	0.00	2.0 MB	Jā
5467122c90cac	Pabeigts	0.00	2.0 MB	Jā

Mazākais skaitlis

Apraksts

Uzrakstiet programmu, kas ievadītam naturālam skaitlim n nosaka, kādu mazāko naturālo skaitli var izveidot no skaitļa n cipariem, katru no tiem izmantojot vienu reizi. Jaunizveidotā skaitļa pieraksts nedrīkst sākties ar 0.

Risinājums

```
#include <iostream>

using namespace std;

const int MAX = 1010;

int n, m, k;
int laukums[MAX][MAX];

void minas(int a, int b) {
    for (int i = -1; i <= 1; i++) {
        for (int j = -1; j <= 1; j++) {
            int rinda = a + i, kol = b + j;
            if (rinda >= 0 && rinda < n && kol >= 0 && kol < m && laukums[rinda][kol] != -1) {
                laukums[rinda][kol]++;
            }
        }
    }
}

int main() {
    cin >> n >> m >> k;
    while (k--) {
        int a, b;
        cin >> a >> b;
        a--;
        b--;
        laukums[a][b] = -1;
        minas(a, b);
    }
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            if (laukums[i][j] == -1) {
                cout << "*";
            } else {
                cout << laukums[i][j];
            }
        }
        cout << endl;
    }
    return 0;
}
```

Rezultāts (11/11 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
545456fc9d53d	Pabeigts	0.00	2.0 MB	Jā
545456fc9d5f6	Pabeigts	0.00	2.0 MB	Jā
545456fc9d6ac	Pabeigts	0.00	2.0 MB	Jā
545456fc9d76d	Pabeigts	0.00	2.0 MB	Jā
545456fc9d81c	Pabeigts	0.00	2.0 MB	Jā
545456fc9d8e6	Pabeigts	0.00	2.0 MB	Jā
545456fc9d9a1	Pabeigts	0.00	2.0 MB	Jā
545456fc9da62	Pabeigts	0.00	2.0 MB	Jā
545456fc9db22	Pabeigts	0.00	2.0 MB	Jā
545456fc9dbe0	Pabeigts	0.00	2.0 MB	Jā
545b8881373d3	Pabeigts	0.00	2.0 MB	Jā

Minesweeper

Apraksts

$n \times m$ rūtiņu lielā taisnstūrveida laukumā atrodas k mīnas. Jūsu uzdevums izvadīt $n \times m$ rūtiņu laukumu kur rūtiņā ierakstīts '*' ja rūtiņā atrodas mīna, pretējā gadījumā skaits cik daudz mīnu atrodas 8 blakus esošajās rūtiņās. Garantēts, ka vienā rūtiņā nav vairāk par 1 mīnu.

Risinājums

```
#include <iostream>

using namespace std;

const int MAX = 1010;

int n, m, k;
int laukums[MAX][MAX];

void minas(int a, int b) {
    for (int i = -1; i <= 1; i++) {
        for (int j = -1; j <= 1; j++) {
            int rinda = a + i, kol = b + j;
            if (rinda >= 0 && rinda < n && kol >= 0 && kol < m && laukums[rinda][kol] != -1) {
                laukums[rinda][kol]++;
            }
        }
    }
}

int main() {
    cin >> n >> m >> k;
    while (k--) {
        int a, b;
        cin >> a >> b;
        a--;
        b--;
        laukums[a][b] = -1;
        minas(a, b);
    }
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            if (laukums[i][j] == -1) {
                cout << "*";
            } else {
                cout << laukums[i][j];
            }
        }
        cout << endl;
    }
    return 0;
}
```

Rezultāts (13/13 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5454ad0897ea4	Pabeigts	0.00	2.0 MB	Jā
5454ad0897f57	Pabeigts	0.00	2.0 MB	Jā
5454ad08982f7	Pabeigts	0.00	2.0 MB	Jā
5454ad0898408	Pabeigts	0.00	2.0 MB	Jā
5454ad08984f0	Pabeigts	0.00	2.0 MB	Jā
5454ad089a9a9	Pabeigts	0.00	2.0 MB	Jā
5454ad089aba5	Pabeigts	0.00	2.0 MB	Jā
5454ad089ac82	Pabeigts	0.00	2.0 MB	Jā
5454ad089eaea	Pabeigts	0.00	2.0 MB	Jā
5454ad089ebe6	Pabeigts	0.00	2.0 MB	Jā
5454ad089ecbb	Pabeigts	0.00	2.0 MB	Jā
5454ad08a1fdb	Pabeigts	0.00	2.0 MB	Jā
5454afeff3610	Pabeigts	0.00	2.0 MB	Jā

Mārciņas, šiliņi un pensi

Apraksts

Kādā valstī līdz nesenam laikam apgrozībā esošās naudas vienības bija mārciņas, šiliņi un pensi. Šīs naudas vienības savā starpā saistīja sekojošas sakarības:

1 mārciņa = 20 šiliņi

1 šiliņš = 12 pensi

Tā kā šāda naudas sistēma bija grūti saskaņojama ar citu valstu naudas sistēmām, tad tika nolemts pāriet uz jaunu sistēmu, kurā būtu tikai mārciņas un pensi (sauksim tos par jaunajām mārciņām un jaunajiem pensiem), pie kam būtu spēkā sakarība 1 jaunā mārciņa = 100 jaunie pensi. Tika nolemts, ka pensam sava vērtība jā saglabā (1 jaunais pens = 1 penss).

Uzrakstiet programmu, kas dotai naudas summai vecajā sistēmā aprēķina atbilstošu naudas summu jaunajā sistēmā.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int n1, n2, n3, m=0, p=0;
    cin>>n1>>n2>>n3;

    p = n1*240+n2*12+n3;
    m = p/100;
    p=p%100;

    cout<<m<<" "<<p;
}
```

Rezultāts (12/12 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
54545727190d3	Pabeigts	0.00	2.0 MB	Jā
545457271918f	Pabeigts	0.00	2.0 MB	Jā
5454572719247	Pabeigts	0.00	2.0 MB	Jā
54545727192f5	Pabeigts	0.00	2.0 MB	Jā
54545727193a1	Pabeigts	0.00	2.0 MB	Jā
545457271944c	Pabeigts	0.00	2.0 MB	Jā
5454572719522	Pabeigts	0.00	2.0 MB	Jā
54545727195c0	Pabeigts	0.00	2.0 MB	Jā
545457271967f	Pabeigts	0.00	2.0 MB	Jā
5454572719732	Pabeigts	0.00	2.0 MB	Jā
5466881e9f780	Pabeigts	0.00	2.0 MB	Jā
5466881ea0630	Pabeigts	0.00	2.0 MB	Jā

Pēdējie cipari

Apraksts

Doti divi veseli skaitļi ($0 < a, b < 10000$). Atrast abu skaitļu pēdējo ciparu summu.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int a, b, sum;
    cin>>a;
    cin>>b;

    a=a%10;
    b=b%10;
    sum=a+b;

    cout<<sum;
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
56920941e1c74	Pabeigts	0.00	2.0 MB	Jā
56920965137df	Pabeigts	0.00	2.0 MB	Jā
569209704654f	Pabeigts	0.00	2.0 MB	Jā
5692098693d5a	Pabeigts	0.00	2.0 MB	Jā
569209994e27d	Pabeigts	0.00	2.0 MB	Jā

Reizināšana ar 2

Apraksts

Šis ir ļoti vienkāršs uzdevums. Jūsu programmai ir jānolasa skaitlis N un jāizvada šis skaitlis pareizināts ar 2.

Risinājums

```
#include <iostream>
int main() {
    int n;
    std::cin >> n;
    std::cout<<n*2;
}
```

Rezultāts (5/5 punkti)

	Statuss	Laiks	Atmiņa	Vērtējums
545456fc151cf	Pabeigts	0.00	2.0 MB	Jā
545456fc152c9	Pabeigts	0.00	2.0 MB	Jā
545456fc15384	Pabeigts	0.00	2.0 MB	Jā
545456fc15440	Pabeigts	0.00	2.0 MB	Jā
545456fc15500	Pabeigts	0.00	2.0 MB	Jā

Trīs skaitļu sakārtošana

Apraksts

Uzrakstiet programmu, kas dotos trīs veselos skaitļus izvada nedilstošā secībā!

Risinājums

```
#include <iostream>

using namespace std;

int main() {
    long a, b, c;
    cin >> a >> b >> c;

    // Bubble sort
    if (a > b) {
        long temp = a;
        a = b;
        b = temp;
    }
    if (b > c) {
        long temp = b;
        b = c;
        c = temp;
    }
    if (a > b) {
        long temp = a;
        a = b;
        b = temp;
    }

    cout << a << " " << b << " " << c << endl;

    return 0;
}
```

Rezultāts (32/32 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums	ID	Statuss	Laiks	Atmiņa	Vērtējums
545457cd17ba5	Pabeigts	0.00	2.0 MB	Jā	545457cd18a5e	Pabeigts	0.00	2.0 MB	Jā
545457cd17c6b	Pabeigts	0.00	2.0 MB	Jā	545457cd18b1c	Pabeigts	0.00	2.0 MB	Jā
545457cd17d1f	Pabeigts	0.00	2.0 MB	Jā	545457cd18be3	Pabeigts	0.00	2.0 MB	Jā
545457cd17dce	Pabeigts	0.00	2.0 MB	Jā	545457cd18ca8	Pabeigts	0.00	2.0 MB	Jā
545457cd17e7e	Pabeigts	0.00	2.0 MB	Jā	545457cd18d71	Pabeigts	0.00	2.0 MB	Jā
545457cd17f2d	Pabeigts	0.00	2.0 MB	Jā	545457cd18e27	Pabeigts	0.00	2.0 MB	Jā
545457cd17fdb	Pabeigts	0.00	2.0 MB	Jā	545457cd18edd	Pabeigts	0.00	2.0 MB	Jā
545457cd1808a	Pabeigts	0.00	2.0 MB	Jā	545457cd18f93	Pabeigts	0.00	2.0 MB	Jā
545457cd1813c	Pabeigts	0.00	2.0 MB	Jā	545457cd19050	Pabeigts	0.00	2.0 MB	Jā
545457cd181eb	Pabeigts	0.00	2.0 MB	Jā	545457cd19107	Pabeigts	0.00	2.0 MB	Jā
545457cd1829a	Pabeigts	0.00	2.0 MB	Jā	54670f2f10931	Pabeigts	0.00	2.0 MB	Jā
545457cd18368	Pabeigts	0.00	2.0 MB	Jā	54670fca36ee3	Pabeigts	0.00	2.0 MB	Jā
545457cd18413	Pabeigts	0.00	2.0 MB	Jā					
545457cd184e4	Pabeigts	0.00	2.0 MB	Jā					
545457cd185a6	Pabeigts	0.00	2.0 MB	Jā					
545457cd186a4	Pabeigts	0.00	2.0 MB	Jā					
545457cd18763	Pabeigts	0.00	2.0 MB	Jā					
545457cd18823	Pabeigts	0.00	2.0 MB	Jā					
545457cd188e0	Pabeigts	0.00	2.0 MB	Jā					
545457cd1899f	Pabeigts	0.00	2.0 MB	Jā					

Sazarojums1 iesācējiem

Apraksts

Atrodi lielāko no 2 skaitļiem.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int a, b;
    cin>>a>>b;
    if(a>b){
        cout<<a;
    }else{
        cout<<b;
    }
}
```

Rezultāts (4/4 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
560ed5f07d0b9	Pabeigts	0.00	2.0 MB	Jā
560ed5f07d16a	Pabeigts	0.00	2.0 MB	Jā
560ed5f07d209	Pabeigts	0.00	2.0 MB	Jā
560ed61618163	Pabeigts	0.00	2.0 MB	Jā

Sazarojums2 iesācējiem

Apraksts

Atrodi lielāko no 3 skaitļiem.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int a, b, c;
    cin>>a>>b>>c;

    if(a>=b && a>=c){
        cout<<a;
    }else if(b>=a && b>=c){
        cout<<b;
    }else{
        cout<<c;
    }
}
```

Rezultāts (9/9 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
560ed75669c2f	Pabeigts	0.00	2.0 MB	Jā
560ed75669ce2	Pabeigts	0.00	2.0 MB	Jā
560ed75669d86	Pabeigts	0.00	2.0 MB	Jā
560ed75669e1d	Pabeigts	0.00	2.0 MB	Jā
560ed75669eb4	Pabeigts	0.00	2.0 MB	Jā
560ed75669f4c	Pabeigts	0.00	2.0 MB	Jā
560ed75669fe3	Pabeigts	0.00	2.0 MB	Jā
560ed7566a07c	Pabeigts	0.00	2.0 MB	Jā
560ed7970a004	Pabeigts	0.00	2.0 MB	Jā

Sazarojums3 iesācējiem

Apraksts

Pārbaudīt vai skaitlis atrodas starp diviem dotajiem skaitļiem!

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int a, b, c;
    cin>>a>>b;
    cin>>c;
    if(c>a&& c<b){
        cout<<"ir";
    }else{
        cout<<"nav";
    }
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
563242aeadc43	Pabeigts	0.00	2.0 MB	Jā
563242be790f6	Pabeigts	0.00	2.0 MB	Jā
5632435a17880	Pabeigts	0.00	2.0 MB	Jā
5632436920edd	Pabeigts	0.00	2.0 MB	Jā
5632438142a2a	Pabeigts	0.00	2.0 MB	Jā

Sazarojums4 iesācējiem

Apraksts

Pārbaudīt vai skaitlis ir pāra skaitlis.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int sk;
    cin>>sk;
    if(sk%2==0){
        cout<<"ir";
    }else{
        cout<<"nav";
    }
}
```

Rezultāts (5/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5633524d2771f	Pabeigts	0.00	2.0 MB	Jā
56335259e2a88	Pabeigts	0.00	2.0 MB	Jā
563352655c457	Pabeigts	0.00	2.0 MB	Jā
5633526d53de4	Pabeigts	0.00	2.0 MB	Jā
5633527c39ae4	Pabeigts	0.00	2.0 MB	Jā

Tuvākais skaitlis

Apraksts

Ievadītam naturālam skaitlim n un ciparam c atrast tādu mazāko naturālo skaitli k , kuram vienlaicīgi izpildās sekojošas īpašības:

1) $k > n$

2) k pieraksts satur ciparu c .

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int n, c, k;
    cin >> n >> c;

    k = n + 1;
    while (k <= 1000000) {
        int pag = k;
        while (pag != 0) {
            if (pag % 10 == c) {
                cout << k;
                return 0;
            }
            pag /= 10;
        }
        k++;
    }
    return 0;
}
```

Rezultāts (12/12 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5454572ae507f	Pabeigts	0.00	2.0 MB	Jā
5454572ae5143	Pabeigts	0.00	2.0 MB	Jā
5454572ae51fb	Pabeigts	0.00	2.0 MB	Jā
5454572ae52a8	Pabeigts	0.00	2.0 MB	Jā
5454572ae5353	Pabeigts	0.00	2.0 MB	Jā
5454572ae53fe	Pabeigts	0.00	2.0 MB	Jā
5454572ae54ab	Pabeigts	0.00	2.0 MB	Jā
5454572ae5557	Pabeigts	0.00	2.0 MB	Jā
5454572ae5606	Pabeigts	0.00	2.0 MB	Jā
5454572ae56d8	Pabeigts	0.00	2.0 MB	Jā
5460d421014d1	Pabeigts	0.00	2.0 MB	Jā
5460d4210250d	Pabeigts	0.00	2.0 MB	Jā

Summa

Apraksts

Uz papīra lapas stabiņā tika uzrakstīti n naturāli n-ciparu skaitļi ar sekojošām īpašībām:

a) pirmajā (augšējā) uzrakstītajā skaitlī visi cipari ir lielāki par 0;

b) katru nākošo skaitli ieguva no iepriekšējā, tā pirmo ciparu pārnesot uz skaitļa beigām un pārējos ciparus atstājot iepriekšējā secībā.

Tā piemēram, ja pirmais skaitlis stabiņā ir 1234, tad viss skaitļu stabiņš ir šāds:

1234

2341

3412

4123

Uzrakstiet programmu, kas dotam stabiņa augšējam skaitlim nosaka un izvada visu stabiņā uzrakstīto skaitļu summu!

Risinājums

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

int main() {
    int n;
    cin >> n;

    int cipari = log10(n) + 1; // noskaidro cik ciparu ir skaitlī n
    int summa = 0;
    int pag = n;

    for(int i = 0; i < cipari; i++) {
        int pedejais = pag % 10;
        pag /= 10;
        pag += pow(10, cipari - 1) * pedejais; //parviesto pedejo ciparu uz skaitla sakumu un
        //pārējos ciparus pa vienu poz uz kreisi
        summa += pag;
    }
    cout << summa;
}
```

Rezultāts (11/11punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5454572d00460	Pabeigts	0.00	2.0 MB	Jā
5454572d004d5	Pabeigts	0.00	2.0 MB	Jā
5454572d00574	Pabeigts	0.00	2.0 MB	Jā
5454572d0060d	Pabeigts	0.00	2.0 MB	Jā
5454572d006a0	Pabeigts	0.00	2.0 MB	Jā
5454572d00731	Pabeigts	0.00	2.0 MB	Jā
5454572d007b8	Pabeigts	0.00	2.0 MB	Jā
5454572d0082a	Pabeigts	0.00	2.0 MB	Jā
5454572d0089d	Pabeigts	0.00	2.0 MB	Jā
545456fc9dbe0	Pabeigts	0.00	2.0 MB	Jā
5454572d00910	Pabeigts	0.00	2.0 MB	Jā

Saskaitīšana stabiņā

Apraksts

Uz papīra lapas tika uzrakstīts naturāls skaitlis $a_n a_{n-1} a_{n-2} \dots a_2 a_1$, kura visi cipari bija lielāki par 0. Skaitlim bija ne vairāk kā 100 cipari. Pēc tam zem šī skaitļa stabiņā tika uzrakstīti skaitļi $a_{n-1} a_{n-2} \dots a_2 a_1$, $a_{n-2} \dots a_2 a_1, \dots, a_2 a_1, a_1$. Beigās visi skaitļi tika saskaitīti. Piemērs, kad sākotnējais skaitlis ir 7231493, parādīts zīmējumā.

7231493

231493

31493

1493

493

93

3

7496561

Uzrakstiet programmu, kas ievadītam sākotnējam skaitlim izvada aprēķināto summu.

Risinājums

```
#include <iostream>
#include <string>
#include <sstream>

using namespace std;
int main() {
    string str;
    long sk;
    do{
        cin>>str;
    }while(str.length()>100);

    long garums = str.length();
    long skaitli[garums];
    for(long i=0; i<garums; i++){
        stringstream ss(str.substr(i, garums));
        ss>>sk;
        skaitli[i]=sk;
    }

    long sum=0;
    for(long i=0; i<garums; i++){
        sum+=skaitli[i];
    }

    cout<<sum;
}
```

Rezultāts (11/11 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
545456fc2a2b6	Pabeigts	0.00	2.0 MB	Nē
545456fc2a38e	Pabeigts	0.00	2.0 MB	Jā
545456fc2a440	Pabeigts	0.00	2.0 MB	Jā
545456fc2a4ee	Pabeigts	0.00	2.0 MB	Jā
545456fc2a59c	Pabeigts	0.00	2.0 MB	Jā
545456fc2a653	Pabeigts	0.00	2.0 MB	Jā
545456fc2a703	Pabeigts	0.00	2.0 MB	Nē
545456fc2a7c1	Pabeigts	0.00	2.0 MB	Nē
545456fc2a880	Pabeigts	0.00	2.0 MB	Nē
545893cce0999	Pabeigts	0.00	2.0 MB	Jā
545456fc2a1fe	Pabeigts	0.00	2.0 MB	Jā

Saskaitīšana kvadrātā

Apraksts

Dotas divas matricas - A un B. Saskaitiet tās.

Risinājums

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

int main() {
    int n, m;
    cin >> n >> m;

    int A[n][m], B[n][m];
    for (int i = 0; i < 2*n; i++) {
        for (int j = 0; j < m; j++) {
            if (i < n) {
                cin >> A[i][j];
            } else {
                cin >> B[i-n][j];
            }
        }
    }

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            cout << A[i][j] + B[i][j] << " ";
        }
        cout << endl;
    }

    return 0;
}
```

Rezultāts (20/20 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5bd21e470789f	Pabeigts	0.01	2.0 MB	Jā
5bd21e4707a7f	Pabeigts	0.00	2.0 MB	Jā
5bd21e4707b6d	Pabeigts	0.01	2.0 MB	Jā
5bd21e4707dd1	Pabeigts	0.01	2.0 MB	Jā
5bd21e4707f87	Pabeigts	0.00	2.0 MB	Jā
5bd21e4708071	Pabeigts	0.00	2.0 MB	Jā
5bd21e470819b	Pabeigts	0.00	2.0 MB	Jā
5bd21e47082b5	Pabeigts	0.00	2.0 MB	Jā
5bd21e4708413	Pabeigts	0.00	2.0 MB	Jā
5bd21e47084ab	Pabeigts	0.00	2.0 MB	Jā
5bd21e47085ad	Pabeigts	0.00	2.0 MB	Jā
5bd21e470862e	Pabeigts	0.00	2.0 MB	Jā
5bd21e470870e	Pabeigts	0.01	2.0 MB	Jā
5bd21e4708a00	Pabeigts	0.00	2.0 MB	Jā
5bd21e4708bba	Pabeigts	0.00	2.0 MB	Jā
5bd21e4708c4c	Pabeigts	0.00	2.0 MB	Jā
5bd21e4708d2d	Pabeigts	0.00	2.0 MB	Jā
5bd21e4708e38	Pabeigts	0.01	2.0 MB	Jā
5bd21e4708ff9	Pabeigts	0.01	2.0 MB	Jā
5bd21e81be385	Pabeigts	0.00	2.0 MB	Jā

Trīs skaitļi

Apraksts

Uzrakstīt programmu, kas nosaka, vai starp ievadītajiem trijiem veseliem skaitļiem var izvēlēties divus tā, lai to summa būtu

pozitīvs skaitlis

nulle

negatīvs skaitlis

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    long a, b, c;
    cin >> a >> b >> c;

    if(a+b > 0 || a+c>0 || b+c >0) cout<<"VAR\n";
    else cout<<"NEVAR\n";
    if(a+b==0 || a+c==0 || b+c==0) cout<<"VAR\n";
    else cout<<"NEVAR\n";
    if(a+b<0 || a+c<0 || b+c<0) cout<<"VAR\n";
    else cout<<"NEVAR\n";

    return 0;
}
```

Rezultāts (12/12 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5454571543860	Pabeigts	0.00	2.0 MB	Jā
5454571543991	Pabeigts	0.00	2.0 MB	Jā
5454571543a85	Pabeigts	0.00	2.0 MB	Jā
5454571543b91	Pabeigts	0.00	2.0 MB	Jā
5454571543cc0	Pabeigts	0.00	2.0 MB	Jā
5454571543dac	Pabeigts	0.00	2.0 MB	Jā
5454571543e70	Pabeigts	0.00	2.0 MB	Jā
5454571543f3e	Pabeigts	0.00	2.0 MB	Jā
5454571543ffb	Pabeigts	0.00	2.0 MB	Jā
54545715440b7	Pabeigts	0.00	2.0 MB	Jā
54987c103ca2c	Pabeigts	0.00	2.0 MB	Jā
54987c103d78c	Pabeigts	0.00	2.0 MB	Jā

Vidējais skaitlis

Apraksts

Tiek ievadīts nepāra skaits dažādu naturālu skaitļu. Noteikt, kurš skaitlis pēc visu skaitļu sakārtošanas augošā secībā šajā virknē ir vidējais.

Risinājums

```
#include <iostream>
#include <algorithm>

using namespace std;

int main() {
    int n;
    cin >> n;

    int a[n];
    for (int i = 0; i < n; i++) {
        cin >> a[i];
    }

    sort(a, a + n);
    cout << a[n / 2] << endl;

    return 0;
}
```

Rezultāts (11/11 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5454570668e21	Pabeigts	0.00	2.0 MB	Jā
5454570668ef3	Pabeigts	0.01	2.0 MB	Jā
5454570669726	Pabeigts	0.00	2.0 MB	Jā
545457066982e	Pabeigts	0.00	2.0 MB	Jā
5454570669923	Pabeigts	0.00	2.0 MB	Jā
54545706699f5	Pabeigts	0.00	2.0 MB	Jā
5454570669b36	Pabeigts	0.00	2.0 MB	Jā
5454570669c3b	Pabeigts	0.00	2.0 MB	Jā
5454570669dea	Pabeigts	0.00	2.0 MB	Jā
545457066a02a	Pabeigts	0.00	2.0 MB	Jā
5480ad46a03c2	Pabeigts	0.00	2.0 MB	Jā

Reizināšana ar 2 (darbs ar failiem)

Apraksts

Šis ir ļoti vienkāršs uzdevums. Jūsu programmai ir jānolasa skaitlis no ievaddatu faila reiz.in un jāizvada šis skaitlis pareizināts ar 2 failā reiz.out.

Risinājums

```
#include <fstream>
#include <iostream>
using namespace std;
int main() {
    ifstream ievades_fails;
    ofstream izvades_fails;

    ievades_fails.open("reiz.in");
    izvades_fails.open("reiz.out");

    cout<< "Ievadi skaitli\n";
    int N;
    ievades_fails >> N;
    izvades_fails << N*2;
    ievades_fails.close();
    izvades_fails.close();
}
```

Rezultāts (5/5 punkti)

Tests	Rezultāts
1	Pareizs
2	Pareizs
3	Pareizs
4	Pareizs
5	Pareizs

Virkne

Apraksts

No naturālo skaitļu virknes tika izsvīroti visi tie skaitļi, kas dalās bez atlikuma ar kādu no naturāliem skaitļiem a , b vai c ($a, b, c > 1$). Uzrakstiet programmu, kas dotām naturālu skaitļu a, b, c un n vērtībām nosaka, kāds bija n -tais nenosvīrotais virknes skaitlis.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int a, b, c, n;
    cin >> a >> b >> c >> n;

    int skaitlis = 1;
    int nenosvitrotais = 0;

    while (n > 0) {
        bool dalasBezAtlikuma = false;

        if (skaitlis % a == 0 || skaitlis % b == 0 || skaitlis % c == 0) {
            dalasBezAtlikuma = true;
        }

        if (!dalasBezAtlikuma) {
            n--;
            nenosvitrotais = skaitlis;
        }

        skaitlis++;
    }

    cout << nenosvitrotais << endl;

    return 0;
}
```

Rezultāts (8/11 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5454575b0514e	Pabeigts	0.00	2.0 MB	Jā
5454575b05228	Pabeigts	-	2.0 MB	Nē
5454575b0530b	Pabeigts	0.00	2.0 MB	Jā
5454575b053be	Pabeigts	0.00	2.0 MB	Jā
5454575b0546f	Pabeigts	0.00	2.0 MB	Jā
5454575b05520	Pabeigts	0.01	2.0 MB	Jā
5454575b055e4	Pabeigts	0.03	2.0 MB	Jā
5454575b05693	Pabeigts	0.14	2.0 MB	Jā
5454575b05752	Pabeigts	-	2.0 MB	Nē
5454575b0580b	Pabeigts	-	2.0 MB	Nē
5454ac8506dbf	Pabeigts	0.00	2.0 MB	Jā

Virkne

Apraksts

Gadās, ka datora klaviatūrai kāds no taustiņiem iesprūst un tad viena burta vietā tekstā būs vairāki vienādi, pēc kārtas esoši burti. Piemēram, vārda "klavieres" vietā var iznākt "kkkkkkklaavierrrrrees".

Uzrakstiet programmu, kas labo šādas kļūdas - t.i., ievadītā simbolu virknē atrod visas vietas, kurās ir vairāki vienādi simboli pēc kārtas, un katrā šādā vietā atstāj tikai vienu no šiem simboliem, bet pārējos izdzēš un virkni "sabīda kopā".

Risinājums

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

int main() {
    string ievade;
    cin >> ievade;

    string txt;
    char prevChar = '\\0';

    for (int i = 0; i < ievade.length(); i++) {
        char currChar = ievade[i];
        if (currChar != prevChar) {
            txt += currChar;
            prevChar = currChar;
        }
    }
    cout << txt;
}
```

Rezultāts (11/11 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
54545706a8eaa	Pabeigts	0.00	2.0 MB	Jā
54545706a8f69	Pabeigts	0.00	2.0 MB	Jā
54545706a9042	Pabeigts	0.00	2.0 MB	Jā
54545706a9100	Pabeigts	0.00	2.0 MB	Jā
54545706a91bc	Pabeigts	0.00	2.0 MB	Jā
54545706a9282	Pabeigts	0.00	2.0 MB	Jā
54545706a9339	Pabeigts	0.00	2.0 MB	Jā
54545706a93fc	Pabeigts	0.00	2.0 MB	Jā
54545706a94e5	Pabeigts	0.00	2.0 MB	Jā
54545706a95ff	Pabeigts	0.00	2.0 MB	Jā
546632c50963a	Pabeigts	0.00	2.0 MB	Jā

Skaitļu virkne

Apraksts

Apskatām skaitļu virkni, kurai $a[1]=A$ un kuru definē sekojoši $a[i+1]=a[i]+1/a[i]$, ja $i>0$. Uzrakstiet programmu, kas ievadītiem naturāliem skaitļiem A un N noskaidro $a[N]$ visus ciparus pirms komata.

Risinājums

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int A, N;
```

```
    cin >> A >> N;
```

```
    double currSk = A;
```

```
    for (int i = 1; i < N; i++) {
```

```
        currSk = currSk + 1 / currSk;
```

```
    }
```

```
    int sk = static_cast<int>(currSk);
```

```
    cout << sk;
```

```
}
```

Rezultāts (10/10 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
545456fda2f52	Pabeigts	0.01	2.0 MB	Jā
545456fda303e	Pabeigts	0.00	2.0 MB	Jā
545456fda30d7	Pabeigts	0.00	2.0 MB	Jā
545456fda3195	Pabeigts	0.00	2.0 MB	Jā
545456fda3246	Pabeigts	0.00	2.0 MB	Jā
545456fda32f3	Pabeigts	0.00	2.0 MB	Jā
545456fda33a0	Pabeigts	0.00	2.0 MB	Jā
545456fda344a	Pabeigts	0.00	2.0 MB	Jā
545456fda34fa	Pabeigts	0.00	2.0 MB	Jā
545e839a90de5	Pabeigts	0.00	2.0 MB	Jā

Uzdevumi par doto skaitļu virkni

Apraksts

Dota desmit skaitļu virkne. Virknes elementi ir veseli skaitļi robežās no 0 līdz 100. Uzdevums atrast:

pirmā un pēdējā elementa summu;

elementu skaitu, kas lielāki par pirmo elementu;

elementu skaitu, kas mazāki par pēdējo elementu.

Risinājums

```
#include <iostream>
using namespace std;
int main() {
    int arr[10];
    int sk;
    for(int i=0; i<10; i++){
        cin>>sk;
        arr[i]=sk;
    }

    int sum = arr[0]+arr[9];

    int lielsk=0;
    int mazsk=0;
    for(int i=0; i<10; i++){
        if(arr[i]>arr[0]) lielsk++;
        else if(arr[i]<arr[9]) mazsk++;
    }
    cout<<sum<<" "<<lielsk<<" "<<mazsk;
}
```

Rezultāts (4/5 punkti)

ID	Statuss	Laiks	Atmiņa	Vērtējums
5627fc52e7fa5	Pabeigts	0.00	2.0 MB	Jā
5627fcbdec2bc	Pabeigts	0.00	2.0 MB	Jā
5627fcf9602e8	Pabeigts	0.00	2.0 MB	Nē
5627fd48bae29	Pabeigts	0.00	2.0 MB	Jā
5627fc9e6c603	Pabeigts	0.00	2.0 MB	Jā

Virkne (darbs ar failiem)

Apraksts

Gadās, ka datora klaviatūrai kāds no taustiņiem iesprūst un tad viena burta vietā tekstā būs vairāki vienādi, pēc kārtas esoši burti. Piemēram, vārda "klavieres" vietā var iznākt "kkkkkklaavierrrrrees".

Uzrakstiet programmu, kas labo šādas kļūdas - t.i., ievadītā simbolu virknē atrod visas vietas, kurās ir vairāki vienādi simboli pēc kārtas, un katrā šādā vietā atstāj tikai vienu no šiem simboliem, bet pārējos izdzēš un virkni "sabīda kopā".

Risinājums

```
#include <iostream>
#include <fstream>

using namespace std;

int main() {
    ifstream inputFile("virkne3.in");
    ofstream outputFile("virkne3.out");

    char input[251];
    inputFile.getline(input, 251);

    char txt[251];
    char prevChar = '\\0';
    int index = 0;

    for (int i = 0; input[i] != '\\0'; i++) {
        if (input[i] != prevChar) {
            txt[index++] = input[i];
            prevChar = input[i];
        }
    }

    txt[index] = '\\0';

    outputFile << txt << endl;

    inputFile.close();
    outputFile.close();
}
```

Rezultāts (10/10 punkti)

Tests	Rezultāts
1	Pareizs
2	Pareizs
3	Pareizs
4	Pareizs
5	Pareizs
6	Pareizs
7	Pareizs
8	Pareizs
9	Pareizs
10	Pareizs

Skaitļu kārtošana (darbs ar failiem)

Apraksts

Jūsu uzdevums ir sakārtot dotus veselus skaitļus augošā secībā.

Ievaddati

Teksta faila kartosan.dat pirmajā rindā dots skaitļu skaits N ($1 \leq N \leq 100000$). Nākamajās N faila rindās doti kārtojamie skaitļi, pa vienam skaitlim katrā rindā. Visi skaitļi ir robežās no -2 000 000 000 līdz 2 000 000 000 (ieskaitot).

Izvaddati

Teksta faila kartosan.rez vienīgajā rindā jāizvada dotie skaitļi augošā secībā, pa vienam skaitlim katrā rindā.

Risinājums

```
#include <algorithm>
#include <fstream>
#include <iostream>

using namespace std;

int main() {
    ifstream inputFile("kartosan.dat");
    ofstream outputFile("kartosan.rez");

    int N;
    inputFile >> N;

    int sk[N];
    for (int i = 0; i < N; i++) {
        inputFile >> sk[i];
    }
    sort(sk, sk + N);

    for (int i = 0; i < N; i++) {
        outputFile << sk[i] << endl;
    }

    inputFile.close();
    outputFile.close();
}
```

Rezultāts (10/10 punkti)

Tests	Rezultāts
1	Pareizs
2	Pareizs
3	Pareizs
4	Pareizs
5	Pareizs
6	Pareizs
7	Pareizs
8	Pareizs
9	Pareizs