



CONCURS de programare

Ediția a patra a concursului de programare organizat de Gazeta de Informatică începe pe data de 14 octombrie 2002. Pentru detalii referitoare la acest concurs citiți paginile următoare sau vizitați secțiunea de pe site-ul nostru dedicată concursului: www.ginfo.ro/concurs.

Informații generale

Concursul *Bursele Agora* este o competiție internațională destinată elevilor și studenților. Acest concurs este organizat de redacția **GInfo** și editura **Agora Media**.

Vor fi douăzeci de runde, la fiecare dintre ele participanții trebuie să rezolve câte o problemă și să trimită soluțiile prin *e-mail*; adresa *e-mail* a concursului nostru este concurs@ginfo.ro. După cele douăzeci de etape, care se vor desfășura prin *e-mail*, va fi o etapă finală care va avea loc la *Cluj-Napoca*; doar primii zece clasări în urma celor douăzeci de runde vor participa la faza finală.

Problemele care trebuie rezolvate sunt propuse de redacția **GInfo** în colaborare cu studenți și profesori care se ocupă de pregătirea elevilor români care participă la diferite concursuri naționale și internaționale.

Înscrierea

Pentru a participa la această ediție a concursului de programare *Bursele Agora* va trebui, mai întâi, să vă înscrieți. Cea mai simplă modalitate de înscriere este completarea *online* a formularului de înscriere (www.ginfo.ro/concurs/inscriere.shtml) care va fi trimis automat, prin intermediul clientului dumneavoastră de *e-mail*, la concurs@ginfo.ro.

O altă posibilitate o constituie trimiterea unui *e-mail* la adresa concurs@ginfo.ro. Acesta va trebui să conțină informațiile cerute în formularul disponibil *online* și anume: numele și prenumele dumneavoastră, data nașterii, adresa *e-mail*, adresa poștală și numărul de telefon.

Cea de-a treia posibilitate constă în trimiterea unei scrisori care va conține aceleași informații. Aceasta trebuie trimisă pe adresa:

Redacția GInfo

str. Universității nr. 8/8, 3400 Cluj-Napoca

În termen de cinci zile de la primirea de către redacția **GInfo** a *e-mail*-ului sau a scrisorii dumneavoastră, vă vom trimite un *e-mail* care va conține codul dumneavoastră de identificare pentru ediția 2002/2003 a concursului.

Desfășurare concursului

Limbajele de programare acceptate la concursul *Bursele Agora* sunt *Pascal* și *C/C++*. Fișierele sursă vor fi compilate folosind *Borland Pascal 7.0* sau *Borland C++ 3.1*.

Problemele vor fi publicate pe *site-ul Web* al revistei noastre într-o secțiune dedicată acestui scop (www.ginfo.ro/concurs/probleme.shtml). Rezolvările vor trebui trimise înainte expirării termenului limită specificat pentru fiecare rundă.

O singură soluție va fi acceptată pentru fiecare problemă. Dacă un concurent va trimite mai multe soluții pentru o anumită problemă, numai prima va fi luată în considerare, restul fiind ignorate. Redacția **GInfo** își rezervă dreptul de a descalifica concurenții care trimit soluții identice (sau aproape identice).

Este interzisă folosirea de secvențe scrise în limbaj de asamblare; dacă un fișier sursă conține astfel de secvențe concurentul va primi 0 puncte pentru problema de la runda respectivă.

Este interzis ca programele generate folosind sursele concurenților să modifice *timer*-ul. Concurenții care vor încerca să modifice ceasul sistem vor fi descalificați.

Dacă aveți neclarități referitoare la enunțurile problemelor, puteți adresa întrebări folosind adresa *e-mail* a concursului. Nu puteți adresa decât întrebări ale căror răspunsuri posibile sunt **DA** sau **NU**. Dacă răspunsul poate fi găsit în enunțul problemei sau întrebarea nu respectă condiția anterioară, atunci răspunsul la întrebare va fi **FĂRĂ COMMENTARII**. În cazul în care redacția **GInfo** consideră că răspunsul la o anumită întrebare trebuie primit de toți participanții, atunci cel care a adresat întrebarea va primi răspunsul **ANUNȚ GENERAL**. Anunțurile generale vor fi disponibile *online* în secțiunea dedicată problemelor.

Fiecare etapă se va desfășura pe parcursul unei săptămâni. Enunțul va fi disponibil *online* în ziua de **luni** a săptămânii, pot fi adresate întrebări până **marți**, toate anunțurile generale vor fi disponibile în cursul zilei de **miercuri**, soluțiile trebuie trimise până **vineri**, vor fi corectate **sâmbătă**, iar rezultatele vor fi publicate în cursul **duminicii**.

Trimiterea soluțiilor

Pentru a trimite soluțiile de la o anumită rundă va trebui să trimiteți un *e-mail* având ca subiect codul dumneavoastră de identificare urmat de textul **Runda XX**, unde **XX** reprezintă numărul rundei (01-20). Acest *e-mail* va avea atașată o arhivă **.ZIP**, **.ACE** sau **.RAR** numită **YYYYRXX.EXT** unde



YYYY este codul dumneavoastră de identificare, XX este numărul runde, iar EXT este extensia (RAR, ACE sau ZIP).

Arhiva va conține două fișiere. Primul dintre ele va fi un fișier .BAT care va conține linia de comandă pe care doriți să o folosim pentru a compila fișierul sursă (al doilea fișier al arhivei), corespunzător runde respective. Acest prim fișier al arhivei se va numi **YYYYYRXX.BAT** unde YYYYY este codul dumneavoastră de identificare, iar XX este numărul runde. Cel de-al doilea fișier din arhivă va conține codul sursă corespunzător rezolvării pe care doriți să o trimiteți pentru problema de la runda respectivă. Denumirea acestuia va fi **YYYYYRXX.PAS**, **YYYYYRXX.C** sau **YYYYYRXX.CPP**. Din motive de securitate, arhivele autoex-pandabile nu sunt acceptate.

Arhiva nu va conține nici un alt fișier sau director, deci nu puteți trimite fișiere auxiliare împreună cu fișierele sursă. Dacă programul dumneavoastră trebuie să creeze fișiere auxiliare, atunci acestea vor avea extensia .TMP, vor fi create în directorul curent, și trebuie șterse înaintea terminării execuției programului. Dacă această regulă nu este respectată concurentul va primi 0 puncte pentru problema corespunzătoare, chiar dacă soluția este corectă. Dacă redacția **GInfo** consideră că un concurent pune în pericol buna desfășurare a competiției (prin suprascrierea unor fișiere, crearea de fișiere foarte mari etc.), atunci acel concurent poate fi descalificat.

Toate fișierele trimise devin proprietatea editurii **Agora Media** și nu pot fi folosite în scopuri comerciale fără acordul scris al editurii.

Evaluarea

Pentru fiecare problemă puteți primi maximum 100 de puncte; numărul maxim de puncte pentru întreg concursul este 2000. Pentru corectarea fiecărei probleme vom folosi între 5 și 20 de seturi de date; dacă programul va genera un fișier de ieșire corect pentru un anumit set de date, atunci concurentul va primi un număr de puncte care se va adăuga la numărul de puncte corespunzător problemei respective.

Evaluarea va fi automată, în vederea efectuării acesteia se va folosi întotdeauna ultima versiune a programului de evaluare **GInfo Evaluator**.

Calculatorul folosit pentru evaluare are următoarea configurație: procesor **AMD Athlon XP 1600+** (1,4 GHz), RAM: 128 MB, HDD: 13 GB.

Este garantat faptul că în timpul evaluării vor fi liberi cel puțin 400 KB de memorie convențională și cel puțin 100 MB de spațiu pe disc. Sistemul de operare folosit în timpul evaluării va fi **Microsoft Windows XP Professional**. Dacă un program va efectua o operație ilegală, atunci concurentul va primi 0 puncte pentru setul de date corespunzător.

În cazul în care configurația *hardware* sau *software* a calculatorului folosit pentru evaluare se modifică, un anunț în acest sens va fi disponibil pe pagina principală a site-ului nostru (www.ginfo.ro).

Sfaturi

Programatorii în *Pascal* nu ar trebui să folosească niciodată *unit*-ul CRT, deoarece programele lor se vor opri returnând eroarea de execuție **Error 200: Division by zero**. Acest fapt este cauzat de un bine cunoscut *bug* al mediului de programare **Borland Pascal 7.0**; nici o problemă propusă spre rezolvare în cadrul concursului *Bursele Agora* nu necesită folosirea acestui *unit*.

Atunci când sunt folosite anumite tipuri de date (cum ar fi *Comp*) modul 8087 trebuie să fie activ. Opțiunile de mediu permit setarea acestui mod ca mod implicit, dar în mod normal acest mod nu este activ. Programatorii în *Pascal* sunt rugați să includă în codul sursă directiva de compilare `{ $N+ }` dacă doresc să folosească tipuri de date care necesită activarea acestui mod. În caz contrar, în momentul în care vom încerca să compilăm fișierul sursă, va fi generată eroarea de compilare **Error 116: Must be in 8087 mode to compile this**. Pentru a vă asigura că programul se va compila exact ca pe calculatorul folosit pentru a-l scrie, apăsați **CTRL+O, O**. Astfel, în sursa dumneavoastră, vor fi incluse toate directivele de compilare cu opțiunile care sunt setate în mediu. Puteți modifica aceste opțiuni astfel încât să fie cele dorite de dumneavoastră. Nu vă îngrijorați dacă nu cunoașteți semnificația unor anumite directive de compilare; acestea nu vor avea efecte negative asupra execuției programului - dacă programul dumneavoastră va rula corect pe calculatorul pe care l-ați folosit pentru a-l scrie, atunci va rula corect și pe calculatorul folosit pentru evaluare.

Toate programele trebuie să citească datele de intrare din fișierul de intrare specificat. Nu ar trebui să fie așteptate date de la intrarea standard (tastatură). Dacă programul dumneavoastră va aștepta o reacție de la tastatură, atunci execuția sa va fi oprită după expirarea timpului și nu veți obține nici un punct pentru acea soluție chiar dacă ea este corectă. Deci, vă rugăm să nu includeți în codul sursă secvențe de tipul `Readln`, `getch()`, `getche()` etc.

Toate programele trebuie să scrie datele de ieșire în fișierul de ieșire specificat. Puteți scrie date și la ieșirea standard, dar aceasta nu va duce decât la încetinirea programului, deci nu este recomandabil să scrieți date pe ecran. Toate datele scrise la ieșirea standard sunt ignorate în timpul evaluării. Programul de evaluare va verifica numai soluția scrisă în fișierul de ieșire specificat. Fișierul de intrare se află întotdeauna în directorul curent; nu încercați să citiți date din alte locații. Fișierul de ieșire va trebui generat tot în directorul curent; nu scrieți date de ieșire în alte locații. Trebuie să folosiți întotdeauna denumirile specificate în enunț pentru fișierul sursă, fișierul de intrare și fișierul de ieșire. Folosirea altor denumiri va duce la funcționarea incorectă a programului.

Exemplu

Denumirea fișierului de intrare: **SUMA.IN**

Denumirea fișierului de ieșire: **SUMA.OUT**

Descriere: Dându-se două numere, aflați suma lor.

Date de intrare: Fișierul de intrare conține, pe o singură linie, cele două numere separate printr-un singur spațiu.

Date de ieșire: Fișierul de ieșire ar trebui să conțină suma celor două numere.

Exemplu:

SUMA . IN	SUMA . OUT
2 4	6

Țimp maxim de execuție: 1 secundă/test

După citirea enunțului veți începe să rezolvați problema. Veți observa că nu este clar dacă numerele sunt întregi sau reale.

Pentru a clarifica acest lucru va trebui să adresați o întrebare. Aceasta nu poate fi: *Numerele sunt întregi sau reale?* deoarece nu se poate răspunde la ea prin DA sau NU. O soluție ar fi să întrebați: *Numerele sunt întregi?* Veți primi răspunsul NU și veți ști că numerele sunt reale.

Acum apare o întrebare mult mai importantă: *Ce precizie trebuie folosită pentru afișarea rezultatului?* Se pare că redacția *GInfo* a uitat să precizeze acest lucru în enunț. Bineînțeles, va trebui să adresați o întrebare la care să se poată răspundă prin DA sau NU. Ați putea încerca: *Este suficientă o precizie de trei zecimale?* Răspunsul la întrebare va fi ANUNȚ GENERAL și va fi urmat de publicarea în secțiunea dedicată problemelor (www.ginfo.ro/concurs/probleme.shtml) a unui mesaj de genul:

Rezultatul va trebui scris cu o precizie de două zecimale. Ne cerem scuze pentru descrierea incompletă a problemei.

Sperăm să nu fim nevoiți să publicăm prea des astfel de mesaje, dar... După citirea anunțului general puteți începe scrierea codului sursă. O soluție corectă ar putea fi:

```
1: Program SUMA;
2: var a,b:Real;
3:     f:Text;
4: Begin
5:   Assign(f,'SUMA.IN');
6:   Reset(f);
7:   Readln(f,a,b);
8:   Close(f);
9:   Assign(f,'SUMA.OUT');
10:  Rewrite(f);
11:  Writeln(f,a+b:0:2);
12:  Close(f);
13: End.
```

Numerele de la începutul fiecărei linii nu fac parte din program; ele sunt folosite pentru a ne putea referi mai ușor la o anumită linie.

Dacă veți adăuga o linie 1': `uses Crt;` și o linie 4': `Clrscr;` nu veți reuși să obțineți decât eroarea Error 200: Division by zero așa cum am explicat în secțiunea *Sfaturi*.

Dacă adăugați o linie 12': `Readln;` rezultatul evaluării va fi `Time out!`, deoarece execuția programului nu se

va termina, ci se va aștepta apăsarea tastei <ENTER>. Execuția va fi întreruptă de către programul de evaluare după o secundă.

Dacă folosiți linia 5: `Assign(f,'C:\SUMA.IN');` atunci programul va funcționa corect numai dacă directorul curent este rădăcina *drive*-ului C: (ceea ce este foarte improbabil). Nu veți obține decât eroarea File not found.

Dacă folosiți linia 5: `Assign(f,'ALTFISIER');` atunci programul nu va citi datele de intrare din fișierul specificat și va rula incorect.

Dacă folosiți linia 9: `Assign(f,'C:\SUMA.OUT');` atunci programul va părea că funcționează corect, dar programul de evaluare nu va găsi fișierul dumneavoastră de ieșire decât dacă directorul curent este rădăcina *drive*-ului C: (din nou, șansele sunt foarte mici). Programul de evaluare va returna `No output file!`.

Dacă folosiți linia 9: `Assign(f,'ALTFISIER');` atunci programul nu va scrie datele în fișierul de ieșire specificat, iar programul de evaluare va returna `No output file!`.

Dacă folosiți linia 7: `Readln(a,b);` atunci programul va aștepta datele de intrare de la intrarea standard și va fi întrerupt după o secundă. Puteți folosi această linie numai dacă aveți o secvență asemănătoare cu:

```
5: Assign(Input,'SUMA.IN');
6: Reset(Input);
7: Readln(a,b);
8: Close(Input);
```

Dacă folosiți linia 11: `Writeln(a+b:0:2);` atunci datele de ieșire vor fi scrise pe ecran și nu în fișierul de ieșire specificat. Puteți folosi această linie numai dacă secvența este de tipul:

```
9: Assign(Output,'SUMA.OUT');
10: Rewrite(Output);
11: Writeln(a+b:0:2);
12: Close(Output)
```

Acestea sunt cele mai des întâlnite erori care au apărut la precedentele ediții ale concursului organizat de *GInfo*. Sperăm că nu veți face aceleași greșeli și la această ediție.

După scrierea codului sursă trebuie să trimiteți soluția. În cele ce urmează vom considera că *ID*-ul dumneavoastră este 02457 și participați la prima rundă a concursului. Trebuie să creați arhiva pe care o veți trimite; aceasta se va numi 02457R01.ZIP, 02457R01.ACE sau 02457R01.RAR și va conține două fișiere: 02457R01.BAT și 02457R01.PAS. Fișierul 02457R01.PAS va conține codul sursă al rezolvării, în timp ce fișierul 02457R01.BAT va conține linia de comandă care va fi folosită pentru compilare, de exemplu `bpc 02457R01.PAS`. Folosiți `bpc` pentru compilarea surselor *Pascal* și `bcc` pentru compilarea surselor *C/C++*.

După crearea acestui fișier trebuie construită arhiva pe care o veți trimite. Folosiți un arhivator care creează arhive .RAR, .ACE sau .ZIP; nu creați o arhivă autoexpandabilă. Acum puteți trimite *e-mail*-ul care va conține soluția. Nu uitați să atașați arhiva și să scrieți în linia *Subject*: 02457 Runda 01.

