

**КОНТРОЛНО СЪСТЕЗАНИЕ
ЗА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР
София, 17 юни 2012 г.
Група А**

Задача А1. VILLIE

Автор: Георги Гюрчев

Велислава мечтае да стане програмист. Тя обаче е много претенциозна и не може да си избере език за програмиране, защото винаги намира нещо, което не ѝ харесва. След като разбрала, че едва ли ще намери такъв, в който няма променливи и типове (които толкова много я объркват), Велислава решила да си измисли свой език, който да отговаря на всичките ѝ изисквания. Езикът VILLIE (Variable-free Intelligent Language for Lame Integer Expressions) има следният синтакс:

- Всички типове са цели числа. За **false** се използва 0, а за **true** – всички останали числа.
- Изразите могат да съдържат константи, аргументи на функции, скоби **()**, повикване на функции и операторите **<, >, =, <=, >=, +, -, *, /** (целочислено деление) и **if**. Операторите за сравнение могат да се срещат само веднъж в даден израз без оператор **if** и се използват само за условието на **if** (но не присъстват задължително).
- Условният оператор **if** има следния синтакс: "**if <израз-условие> then <израз1> else <израз2>**". Той може да се използва и по средата на даден израз, но за да се избегнат двусмислия, ще бъде ограден в скоби **()**.
- Функциите се декларират по следния начин: "**fun <име> (<аргументи, разделени със запетаи>) = <израз>;**" – всичко на един ред.
- Имената на аргументите на някоя функция могат да бъдат малки и главни латински букви, а имената на функциите - 2 или повече малки или главни латински букви.
- Една функция може да бъде повикана от която и да е част от програмата, стига да е декларирана някъде.
- Когато на някой ред от програмата е зададен "**<израз>;**", то стойността на израза се изчислява и директно се извежда на стандартния изход на програмата (всеки израз на нов ред).

Имало само един проблем: Велислава няма необходимите познания по други езици, за да напише компилатор. Затова тя се нуждае от вашата помощ. Напишете програма **villie**, която изпълнява дадена програма на езикът VILLIE.

Вход

На всеки ред от входа има декларация на функция или изчисляване на израз. Последният ред също завършва със знак за нов ред.

Изход

Трябва да съдържа изхода на зададената програма.

Ограничения

Програмата няма да съдържа повече от 50 реда. Всеки от тях ще е не по-голям от 150 символа. Дълбочината на рекурсиите е не по-голяма от 500 000. $-2^{31} \leq$ всички междинни резултати $< 2^{31}$. Алгоритмите в програмата няма да превишават сложността по време и памет от зададените ви, т.е. ако програмата бъде директно "преведена" на

C/C++, условията за време и памет ще бъдат спазени (възможно е изключение за стек паметта). Няма да се среща деление на нула и изрази, започващи с - (минус).

Пример

Вход

```
2+4/2-1*(2/2-1);  
fun fact(n)=if n<=1 then 1 else n*fact(n-1);  
fun Sum(N)=if N=1 then 1 else N+Sum(N-1);  
fun func(x,y)=1+(if x+y<3 then 3 else if x=4 then 0 else 1)+x-y;  
fact(4);  
fact(4+6);  
Sum(10);  
fact(Sum(3));  
func(3,0);  
func(3,1);  
func(4,0-1);
```

Изход

```
4  
24  
3628800  
55  
720  
5  
4  
6
```

Оценяване

- В 20% от тестовете няма да има декларации на функции.
- В други 20% няма да има рекурсивни повиквания на функции (но е възможно една функция да използва друга).
- В други 20% дълбочината на рекурсията ще е не повече от 100.