**Анализ на решението на задача**

**Излишни скоби**

Това е задача за логически разсъждения и умело използване на рекурсия. Възможни са решения с изчерпване, но те ще са бавни. В началото на автора му се струваше, че ще се наложи използването на подход, подобен на обратен полски запис, но след по-внимателно обмисляне стана ясно, че е възможно линейно решение, базиращо се на няколко доста прости правила. За да ги формулираме, ще разгледаме всевъзможните случаи на „обкръжение“ на двойка съответни отваряща и затваряща скоби. Под „обкръжение“ тук ще разбираме двата символа, които стоят преди отварящата скоба и след затварящата скоба. За да не отделяме специално внимание на случаите, в които отварящата скоба е първи символ в низа или затварящата е последен символ, нека приемем, че в началото и края на низа, представляващ аритметичния израз, са поставени по един „служебен“ символ, напр. ‘%’. И така, да разгледаме възможните случаи на „обкръжения“ на двойка съответни отваряща и затваряща скоби и да дефинираме в кои случаи двойката скоби е излишна:

1. %(......)символ, раличен от ‘\*’ и ‘/’ : такава двойка скоби винаги е излишна;
2. ((.......)символ, различен от ‘\*’ и ‘/’ : такава двойка скоби винаги е излишна;
3. +(......)символ, различен от ‘\*’ и ‘/’ : такава двойка скоби винаги е излишна;
4. –(......) : такава двойка скоби е излишна тогава и само тогава, когато **вътре** в скобите се срещат само операции \* и /.
5. \*(…..) : такава двойка скоби е излишна тогава и само тогава, когато **вътре** в скобите се срещат само операции \* и /.
6. Символ, различен от ‘/’(.......)\* : такава двойка скоби е излишна тогава и само тогава, когато **вътре** в скобите се срещат само операции \* и /.
7. Символ, различен от ‘/’(……)/ : такава двойка скоби е излишна тогава и само тогава, когато **вътре** в скобите се срещат само операции \* и /.

Да отбележим, че ситуациите %(.....)), ((......)% не могат да се срещнат в правилен аритметичен израз.

Това са всички възможни случаи, в които се появяват излишни двойки скоби.

*Терминът „****вътре*** *в скобите“ означава, че се гледат* ***само*** *операциите, които са обхванати само от двойката скоби, които разглеждаме, а не и от скоби, които са вложени в тях* ***и не са премахнати.*** *Ако премахнем някоя двойка вложени скоби, то операциите, които са обхванати от тях, се „вдигат“ едно ниво нагоре и вече са „вътре“ в двойката обхващащи скоби от по-горното ниво.*

Разполагайки с тези правила, можем да обходим въведения аритметичен израз отляво надясно, като, срещайки отваряща скоба, викаме процедура, която продължава обхода докато срещне съответната затваряща скоба. При този обход, ако срещне нова отваряща скоба тя вика рекурсивно себе си. По време на обхода от отварящата до съответната ѝ затваряща, процедурата следи какви операции са се срещнали между двете скоби (само на тяхното ниво, а не във вложени в тях скоби) и, когато срещне съответната затваряща скоба, проверява „обкръжението“ и какви операции е имало в скобите. Според резултата от проверката, решава дали тази двойка скоби е излишна или не. Може да се използва масив или вектор с дължината на първоначалния низ, който в началото е нулиран. Когато се открие излишна двойка скоби, в съответните позиции на този масив се поставят единици, които са индикация, че тези символи не трябва да бъдат извеждани. След това последователно се извеждат всички символи от низа, които не са маркирани с единица. Решението е линейно по броя символи във входния низ.

*Автор: Руско Шиков*