

КОРЕКТНОСТ НА АЛГОРИТМИ, ОПИСАНИ ЧРЕЗ БЛОКСХЕМИ

Коректност на алгоритми, описани чрез блоксхеми, се доказва с помощта на множество от инварианти — по един инвариант за всеки преход в схемата. Тъй като блоковете на схемата съответстват на командите на алгоритъма, то те предизвикват промени в състоянието на паметта (например промяна на стойността на някоя променлива). Преходите (стрелките) между блоковете съответстват на моментни състояния на паметта и само за тях е възможно да се изкажат твърдения (инварианти).

За да докажем коректността на алгоритъма по този метод, е необходимо да докажем следната импликация за всеки блок:

*Ако е вярно твърдението върху стрелката, по която влизаме в блока,
то е вярно и твърдението върху стрелката, по която излизаме от блока.*

Следва, че ако е вярно твърдението върху входната стрелка на блоксхемата, то е вярно и твърдението върху изходната стрелка. Тези твърдения се отнасят съответно за допустимостта на входните данни и правилността на резултата. С други думи, по този начин се доказва частичната коректност на алгоритъма: ако подадем допустими данни на входа, се получава верен отговор на изхода, при условие че изобщо се получи някакъв отговор.

За да бъде доказана пълната коректност на алгоритъма, е нужно още нещо: трябва да се убедим, че алгоритъмът изобщо връща отговор, тоест че завършва рано или късно. Това не следва от разсъжденията по-горе, а се доказва отделно. Прави се с помощта на полуинварианти — числови величини, които се изменят строго монотонно (само растат или само намаляват). Подходящо ограничение на стойностите на величината отгоре или отдолу води до извода, че тя приема само краен брой стойности, затова алгоритъмът изпълнява краен брой стъпки, тоест завършва непременно.

Пример: Пресмятане на факториела на цяло неотрицателно число.

