# ИНВАРИАНТИ И ПОЛУИНВАРИАНТИ

Инвариантите и полуинвариантите служат за анализ на различни процеси в най-широкия смисъл на понятието — физични процеси, алгоритми, игри и др.

*Инвариант* се нарича израз, чиято стойност не се мени в хода на процеса, въпреки че съставките на израза могат да се променят. Изразът може да бъде логически или числов.

*Полуинвариант* се нарича израз, чиято стойност се изменя монотонно — или само расте, или само намалява. Невинаги се изисква строга монотонност: при някои процеси е допустимо стойността на израза да остава непроменена за една или повече стъпки. Стойността на израза трябва да бъде реално число.

## Приложения в информатиката

В информатиката инвариантите и полуинвариантите служат за доказване на коректността на алгоритми — итеративни и рекурсивни.

При итеративните алгоритми всеки цикъл притежава собствен инвариант и полуинвариант. Инвариантът има логическа стойност — той е съждение, което се отнася за текущите стойности на променливите на алгоритъма, описва постигнатото от цикъла при всяко изпълнение на неговото тяло и е истина при всяка проверка на условието за край на цикъла (обаче може да се нарушава по време на изпълнение на тялото на цикъла). Инвариантът се доказва чрез математическа индукция по поредния номер на проверката за край на цикъла. Полуинвариантът представлява числов израз, който се отнася за текущите стойности на променливите на алгоритъма и описва напредъка на цикъла, като се изменя строго монотонно. Взети заедно, инвариантът и полуинвариантът дават обосновка на коректността на цикъла: чрез полуинварианта се доказва, че цикълът ще завърши рано или късно, а от инварианта, приложен към последната проверка за край на цикъла, се прави извод за неговия резултат.

Има два вида цикли — цикъл по брояч и цикъл по условие. В първия случай полуинвариант винаги е броячът на цикъла. За циклите по условие липсва просто правило.

Рекурсивните алгоритми нямат инварианти. Тяхната коректност се доказва с математическа индукция по подходящ параметър (който може да не съвпада с никой от входните параметри на алгоритъма, а вместо това може да е израз, зависещ от тях). Че конкретен рекурсивен алгоритъм ще завърши, се доказва с полуинвариант, който обикновено съвпада с параметъра на индукцията.

При алгоритмите, които съдържат и цикли, и рекурсия, двата подхода се прилагат едновременно: по един инвариант и полуинвариант за всеки цикъл, а за рекурсията — математическа индукция и още един полуинвариант.

**Задача 1.** Докажете, че следният итеративен алгоритъм с цикъл по брояч връща сбора на числата от входния масив.

```
Сбор(A[1...n]: числов масив)
s ← 0
for k ← 1 to n do
   s ← s + A[k]
return s
```

**Решение:** Полуинвариант на цикъла е броячът k: той нараства с единица (от 1 до $n+1$) след всяко изпълнение на тялото на цикъла. Следователно то се изпълнява точно n пъти.

Инвариант: При всяка проверка за край на цикъла $k \overset{?}{\le} n$ важи равенството $s = A[1]+A[2]+A[3]+...+A[k-1]$.

Доказателство на инварианта: с индукция по поредния номер на проверката.

База: При първата проверка: $k = 1$, лявата страна $s = 0$, дясната страна също е 0, защото съдържа $k - 1 = 0$ събираеми, тоест тя е празен сбор.

Индуктивна стъпка: Нека $s = A[1]+ A[2]+ A[3]+ ... + A[k-1]$ при някоя проверка за край на цикъла, която не е последна. Тогава $k \le n$ и тялото на цикъла се изпълнява още веднъж. Променливата s приема стойност $\widetilde{s} = s+A[k] = A[1]+A[2]+A[3]+...+A[k-1]+A[k]$, след което броячът k се увеличава с единица, тоест новата му стойност е $\widetilde{k} = k+1$, откъдето $k = \widetilde{k}-1$. Като заместим в последното равенство, то приема вида:

$$\widetilde{s} = A[1]+A[2]+A[3]+...+A[\widetilde{k}-1].$$

Това е тъкмо равенството от инварианта, но записано с новите стойности на променливите. Тоест то важи и при следващата проверка за край на цикъла.

С това инвариантът е доказан и може да бъде използван за обосновка на коректността на алгоритъма.

По-горе доказахме, че цикълът ще завърши. При последната проверка на условието за край важи инвариантът, а също и равенството $k = n+1$. Заместваме го в инварианта:

$$s = A[1]+A[2]+A[3]+...+A[n],$$

тоест стойността на s е равна на сбора от всички елементи на масива. После се изпълнява операторът “**return** s”, който връща тъкмо тази стойност — сбора от всички елементи на дадения числов масив $A[1...n]$.

С това е доказана коректността на алгоритъма.

**Задача 2.** Докажете, че следният итеративен алгоритъм с цикъл по условие намира индекса M на числото K в сортирания числов масив A[1...n] (ако то се среща няколко пъти, алгоритъмът намира едно такова място; ако K изобщо не се среща в масива, алгоритъмът връща минус единица).

```
Двоично търсене(A[1...n]: числов масив; K: реално число)
// Масивът A е сортиран: A[1] ≤ A[2] ≤ ... ≤ A[n].
L ← 1
H ← n
while L ≤ H do
   M ← ⌊(L + H) / 2⌋
   if A[M] = K
      return M  // Масивът A съдържа K на позиция № M.
   else if A[M] < K
      L ← M + 1
   else // A[M] > K
      H ← M - 1
return -1  // Масивът A не съдържа числото K.
```

**Решение:** Нито L, нито H е полуинвариант на цикъла: никое от тях не се изменя строго монотонно (всеки път едно от тях запазва стойността си). Полуинвариант на цикъла е разликата $H - L$: тя намалява поне с единица при всяко изпълнение на тялото на цикъла. Тази разлика е цяло число, понеже H и L са цели числа. Не съществува безкрайна строго намаляваща редица от цели неотрицателни числа, следователно разликата $H - L$ рано или късно ще стане отрицателна, а при $H < L$ цикълът приключва. С това доказахме, че алгоритъмът винаги завършва обработката на данните (т.е. не се зацикля).

Инвариант на цикъла: При всяка проверка на условието за край важи неравенството $A[r] \neq K$ за всяко цяло $r$, за което $1 \leq r < L$ или $H < r \leq n$.

Доказателство на инварианта: с математическа индукция по поредния номер на проверката за край на цикъла.

База: При първата проверка на условието за край на цикъла са в сила равенствата $L = 1$ и $H = n$, затова променливата $r$ с квантор за всеобщност пробягва празното множество и инвариантът е тривиално верен.

Индуктивна стъпка: Нека инвариантът е верен при някоя проверка за край, която не е последна. Затова $L \leq H$ и $A[M] \neq K$ (инак текущата проверка би била последна). По-нататък разглеждаме два случая.

Първи случай: $\texttt{A[M]} < \texttt{K}$. Понеже масивът е сортиран (т.е. елементите му са подредени във възходящ ред), то $\texttt{A[r]} < \texttt{K}$, значи $\texttt{A[r]} \neq \texttt{K}$, за всяко $\texttt{r}$, за което $1 \leq \texttt{r} < \texttt{M + 1}$. Тъй като в този случай $\texttt{L}$ приема нова стойност $\texttt{M + 1}$, то можем да запишем получения извод така: $\texttt{A[r]} \neq \texttt{K}$ за всяко цяло число $\texttt{r}$, за което $1 \leq \texttt{r} < \texttt{L}$. Освен това $\texttt{A[r]} \neq \texttt{K}$ за всяко цяло $\texttt{r}$, за което $\texttt{H} < \texttt{r} \leq \texttt{n}$ (съгласно с индуктивното предположение); и $\texttt{H}$ не се мени в тялото на цикъла. Като обединим двете неравенства, стигаме до индуктивното заключение: инвариантът ще бъде верен и при новата проверка за край на цикъла.

Втори случай: $\texttt{A[M]} > \texttt{K}$. Понеже масивът е сортиран (т.е. елементите му са подредени във възходящ ред), то $\texttt{A[r]} < \texttt{K}$, значи $\texttt{A[r]} \neq \texttt{K}$, за всяко $\texttt{r}$, за което $\texttt{M - 1} < \texttt{r} \leq \texttt{n}$. Тъй като в този случай $\texttt{H}$ приема нова стойност $\texttt{M - 1}$, то можем да запишем получения извод така: $\texttt{A[r]} \neq \texttt{K}$ за всяко цяло число $\texttt{r}$, за което $\texttt{H} < \texttt{r} \leq \texttt{n}$. Освен това $\texttt{A[r]} \neq \texttt{K}$ за всяко цяло $\texttt{r}$, за което $1 \leq \texttt{r} < \texttt{L}$ (съгласно с индуктивното предположение); и $\texttt{L}$ не се мени в тялото на цикъла. Като обединим двете неравенства, стигаме до индуктивното заключение: инвариантът ще бъде верен и при новата проверка за край на цикъла.

С това инвариантът е доказан.

Преминаваме към доказателството за коректност на алгоритъма като цяло. Вече доказахме, че той ще завърши, тоест ще върне някаква стойност. Но има само два оператора **`return`**. Затова отново разглеждаме два случая:

— Ако алгоритъмът завърши с оператора "**`return`** `M`" от тялото на цикъла, то върнатата стойност `M` е вярна: на позиция `M` масивът `A` съдържа числото `K`, т.е. $\texttt{A[M]} = \texttt{K}$. Това твърдение следва от последната проверка "**`if`** $\texttt{A[M]} = \texttt{K}$" (без помощта на инварианта).

— Ако алгоритъмът приключи изпълнението си с оператора "**`return`** `-1`" след цикъла, това означава, че при последната проверка неравенството $\texttt{L} \leq \texttt{H}$ не е било в сила, т.е. $\texttt{L} > \texttt{H}$. Прилагаме инварианта към последната проверка за край на цикъла: $\texttt{A[r]} \neq \texttt{K}$ за всяко цяло r, за което $1 \leq \texttt{r} < \texttt{L}$ или $\texttt{H} < \texttt{r} \leq \texttt{n}$. От $\texttt{L} > \texttt{H}$ следва, че обединението на двата интервала, пробягвани от `r`, покрива целия числов интервал от 1 до `n` включително. Иначе казано, $\texttt{A[r]} \neq \texttt{K}$ за всяко цяло r, за което $1 \leq \texttt{r} \leq \texttt{n}$. Тоест числото `K` не се съдържа в масива `A[1...n]`. Именно това е смисълът на върнатата стойност (–1), така че резултатът на алгоритъма е верен и в този случай.

С това е доказана коректността на двоичното търсене. В доказателството съществено се използва предположението, че масивът `A[1...n]` е сортиран. Не е трудно да се докаже с подходящ контрапример, че двоичното търсене не работи коректно върху несортирани масиви.

**Задача 3.** Докажете, че следният алгоритъм, наречен *бързо степенуване*, пресмята степента $q^n$ (като $0^0$ се приема за равно на 1).

```
Бързо степенуване(q ; n)
// Входни данни:
// q е произволно реално число;
// n е цяло неотрицателно число.
1) if n = 0
2)   return 1
3) else if n ≡ 0 (mod 2) // n е четно
4)   t ← Бързо степенуване(q ; n/2)
5)   return t × t
6) else // n е нечетно
7)   t ← Бързо степенуване(q ; (n-1)/2)
8)   return t × t × q
```

**Решение:** Полуинвариант е $n$: намалява при всяко рекурсивно извикване. Ако $n = 0$, алгоритъмът завършва веднага и връща 1. Иначе $n > 0$ и е цяло число. Не съществува безкрайна строго намаляваща редица от цели положителни числа, затова $n$ ще стане отрицателно или нула. При деление със закръгляне надолу $n$ не може да стане отрицателно, затуй ще стане нула и алгоритъмът ще завърши.

Тъй като алгоритъмът е рекурсивен, той няма инвариант. Коректността му се доказва със силна индукция по $n$ (полуинварианта).

База на индукцията: При $n = 0$ алгоритъмът връща 1, което е правилно, защото $q^n = q^0 = 1$ за всяко реално число $q$, включително нулата (по уговорка).

Индуктивна стъпка: Да приемем, че алгоритъмът, извикан с параметри $q$ и $k$, връща $q^k$ за всяко реално число $q$ и за всяко цяло неотрицателно число $k < n$, където $n$ е някое цяло положително число. Ще докажем, че ако бъде извикан с параметри $q$ и $n$, алгоритъмът ще върне $q^n$. Възможни са два случая.

Първи случай: $n$ е четно (положително) число. Тогава $n/2$ е положително цяло число, по-малко от $n$, и от индуктивното предположение следва, че ред № 4 ще присвои на $t$ степента $q^{n/2}$. Ето защо ред № 5 ще върне $t^2 = \left(q^{n/2}\right)^2 = q^n$.

Втори случай: $n$ е нечетно (положително) число. Тогава $(n-1)/2$ е цяло неотрицателно число, по-малко от $n$, и според индуктивното предположение $t = q^{(n-1)/2}$ след ред № 7, а ред № 8 ще върне $t^2 q = \left(q^{(n-1)/2}\right)^2 q = q^n$.

Бързото степенуване използва около $2 \log_2 n$ умножения в най-лошия случай (когато $n$ е с единица по-малко от степен на двойката). Това е много по-малко от $n - 1$ (броя на умноженията според определението за степен).

## Приложения в математиката

В математически задачи (често състезателни) инвариантите се използват в доказателства за невъзможност: че едно разположение на някакви елементи не може да се получи от друго разположение по определени правила. За целта е достатъчно да намерим инвариант — някаква величина, която не се променя при извършване на кой да е от разрешените ходове и има различни стойности за двете разположения. Полуинвариантите също могат да послужат за тази цел: ако полуинвариантът например намалява при всеки ход и крайната стойност е по-голяма от началната, то от началното разположение не може да се получи желаното крайно разположение. При тази употреба полуинвариантите могат да бъдат и нестроги в зависимост от спецификата на задачата. Но по-често полуинвариантите служат за доказване, че даден процес непременно завършва; в такъв случай полуинвариантът трябва да се изменя строго монотонно.

### Задачи за инварианти

**Задача 1** (сп. "Математика", бр. 5/1993 г., задача 64). На черната дъска са написани числата 1001, 1002, 1003, 1004, 1005, 1006. Разрешено е да изберем които и да е две от тях и към едното да прибавим 7, а от другото да извадим 7. Новите две числа записваме на мястото на старите, след което продължаваме по същия начин. Възможно ли е след няколко такива промени на дъската да има шест равни числа?

**Решение:** Описаните действия пораждат отново цели числа и не променят техния сбор 6021 (тоест сборът им е инвариант). Понеже 6021 е нечетно число, то не се дели на 6, следователно не е възможно да се получат шест равни числа.

Следващите три задачи са взети из статията "Инварианти в задачи" от Цветан Павлов (Казанлък), публикувана в бр. 10/1988 г. на сп. "Математика".

**Задача 2.** В Тихия океан на Сиво-кафяво-червения остров живеят три цвята хамелеони: 133 сиви, 155 кафяви и 177 червени. Ако се срещнат два хамелеона от различен цвят, те едновременно променят цвета си в третия възможен. (Ако например се срещнат сив и кафяв хамелеон, те двата стават червени.) Ако се срещнат два едноцветни хамелеона, те запазват цвета си. Възможно ли е да се случи така, че всички хамелеони на острова да станат в един и същ цвят?

**Решение:** Не може. Каквито и промени да настъпват, множеството от остатъците на трите числа при деление на 3 винаги е $\{0\ ;\ 1\ ;\ 2\}$, тоест винаги присъстват трите остатъка.

**Задача 3.** Числова променлива расте или намалява с 10 или 15. Може ли от начална стойност 0 променливата да достигне до стойност: а) 123 ? б) 125 ?

**Решение:** а) Не може. Стойността на променливата винаги се дели на 5.

б) Може: единайсет пъти увеличаваме стойността с 10 и веднъж с 15.

Второто подусловие на задача 3 ни кара да обърнем внимание на следната важна подробност. Ако от инварианта не следва, че желаното преобразуване е невъзможно, то това още не гарантира, че то наистина може да се осъществи! Може да има друг инвариант, който да създава пречка. Ето защо е нужна примерна редица от ходове, която показва как се постига целта.

Строгото доказателство на който и да е инвариант се извършва с помощта на математическа индукция (по броя на ходовете). Така например в задача 3 базата на индукцията е при нула хода: началната стойност на променливата е 0, а числото 0 се дели на 5. Индуктивната стъпка гласи: ако текущата стойност $A$ се дели на 5, то следващата стойност ще бъде някое от числата $A \pm 10$ и $A \pm 15$, а те всички се делят на 5.

В задачите тук често ще пропускаме формалните подробности, за да можем да се съсредоточим върху най-важното — откриването на подходящ инвариант.

**Задача 4.** Квадрат е разделен на 16 квадратчета, разположени в четири реда и четири стълба. Във всяко квадратче е написан знак плюс освен едно, в което е написан знак минус. Разрешено е да променяме едновременно всички знаци в произволно избран ред, стълб или в някой от двата диагонала.. Възможно ли е минусите да изчезнат?

**Решение:** Ако в избран ред, стълб или диагонал има $m$ минуса, то след като променим знаците в него, броят на минусите в същия ред, стълб или диагонал ще стане равен на $4 - m$. Числата $m$ и $4 - m$ имат еднаква четност, следователно общият брой на минусите запазва четността си. Отначало този брой е единица, тоест нечетно число. Ето защо в таблицата винаги ще има нечетен брой минуси. В частност броят им не може да стане нула, защото нулата е четно число. Извод: минусите не могат да изчезнат.

**Задача 5.** Да се реши в цели положителни числа уравнението

$$1! + 2! + 3! + \dots + n! = k^2.$$

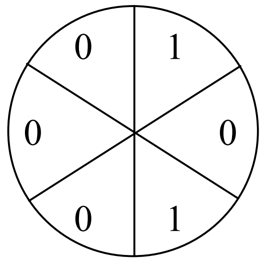
**Решение:** Сред малките стойности на $n$ откриваме две, за които сборът е точен квадрат:

$$1! = 1 = 1^2;$$
$$1! + 2! = 3 \text{ не е точен квадрат;}$$
$$1! + 2! + 3! = 9 = 3^2;$$
$$1! + 2! + 3! + 4! = 33 \text{ не е точен квадрат;}$$
$$1! + 2! + 3! + 4! + 5! = 153 \text{ не е точен квадрат;}$$
$$\text{и тъй нататък.}$$

Намерихме две решения: $n = k = 1$ и $n = k = 3$. Уравнението не притежава никакви други решения, защото при $n \geq 5$ числото $n!$ завършва на цифрата 0, поради което сборът в лявата страна завършва на 3, а никой точен квадрат не завършва на 3 (всички точни квадрати завършват на някоя от цифрите 0, 1, 4, 5, 6 и 9).

Следващите четири задачи са из статията “Инварианти” от Иван Симеонов, публикувана в бр. 1/1995 г. на сп. “Математика плюс”.

**Задача 6.** Кръг е разделен на шест сектора, в които последователно са написани числата 1, 0, 1, 0, 0, 0. Разрешава се едновременното увеличаване с единица на числата в два съседни сектора, избрани произволно. Възможно ли е след няколко такива операции да се получат шест равни числа?




**Решение:** Да означим числата с променливите $a_1$, $a_2$, $a_3$, $a_4$, $a_5$ и $a_6$, например по часовника. Изразът $A = a_1 - a_2 + a_3 - a_4 + a_5 - a_6$ е инвариант, защото при едновременно увеличаване с единица на стойностите на които и да е две съседни променливи (вкл. първата и последната) тази единица хем се прибавя, хем се изважда от $A$. Отначало $A = 1 - 0 + 1 - 0 + 0 - 0 = 2$. Ето защо $A = 2$ след всеки ход. За да се получат шест равни числа, трябва $A$ да стане 0, което е невъзможно. Следователно не може да се получат шест равни числа.

**Задача 7.** Дадена е редицата 1, 2, 3, 4, 5, 1994. На всеки ход избираме две произволни числа от нея, например $a$ и $b$, и заменяме $a$ с $a - 1$ или $a + 1$, а пък $b$ — с $b - 3$ или $b + 3$. Възможно ли е след няколко такива операции да се стигне до редицата 1995, 5, 4, 3, 2, 1 ?

**Решение:** Не може, защото сборът на числата в едната редица е четен, а в другата — нечетен. А пък четността на сбора е инвариант: на всеки ход се променя четността едновременно на две числа от редицата.

**Задача 8.** Дадени са числата 19 и 94. Позволено е да заменим едно от тях със сбора или (абсолютната стойност на) разликата на двете числа. Можем да прилагаме тази операция много пъти. Възможно ли е да получим двойката: а) 19 и 96 ? б) 19 и 95 ?

**Решение:** а) Да, например така:

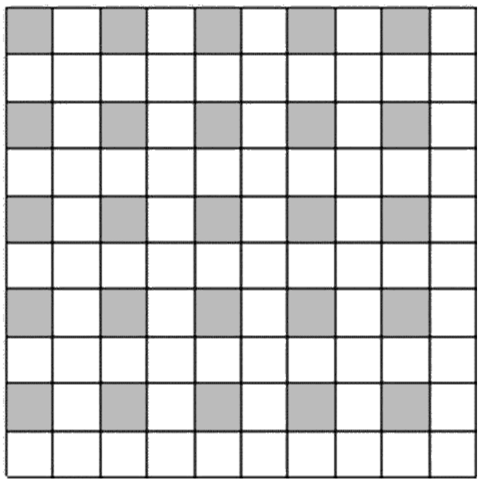
$$\begin{array}{ccccccccccc}
\{19\,;94\} & \to & \{19\,;75\} & \to & \{19\,;56\} & \to & \{19\,;37\} & \to & \{18\,;19\} & \to & \{1\,;18\} \\
 & & & & & & & & & & \downarrow \\
\{19\,;96\} & \leftarrow & \{19\,;77\} & \leftarrow & \{19\,;58\} & \leftarrow & \{19\,;39\} & \leftarrow & \{19\,;20\} & \leftarrow & \{1\,;19\}.
\end{array}$$

б) Не, защото позволените операции запазват най-големия общ делител, а пък $\text{НОД}(19\,;94) = 1 \neq 19 = \text{НОД}(19\,;95)$.

Формално погледнато, оцветяванията като метод за решаване на задачи спадат към инвариантите.

**Задача 9.** Може ли квадрат $10 \times 10$ да се покрие с 25 плочки $4 \times 1$ ?

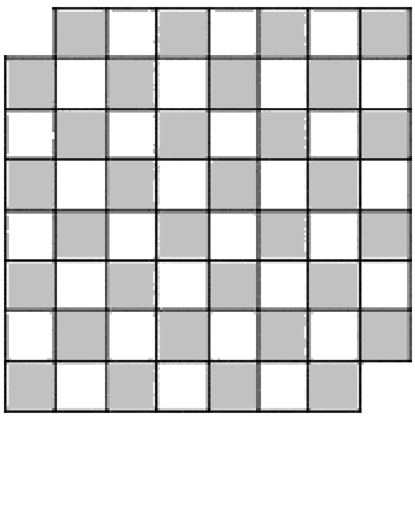
**Решение:** Не може. Разделяме големия квадрат на $10 \times 10 = 100$ квадратчета и ги оцветяваме, както е показано на чертежа.



Всяка плочка $4 \times 1$ покрива четен брой тъмни квадратчета (нула или две), ето защо всичките плочки $4 \times 1$ покриват общо четен брой тъмни квадратчета. Но броят 25 на тъмните квадратчета е нечетно число, следователно те няма как да бъдат покрити с плочки $4 \times 1$.

**Задача 10.** От шахматна дъска $8 \times 8$ са изрязани ъгловите полета a8 и h1. Възможно ли е да се покрие останалата част от дъската с плочки от домино?

**Решение:** Не е възможно, защото при стандартното шахматно оцветяване от дъската остават 30 бели и 32 черни полета след изрязването на a8 и h1, обаче всяка плочка от домино покрива едно бяло и едно черно поле, затова произволен брой плочки за домино ще покрият равен брой бели и черни полета, а не 30 бели и 32 черни.



Инварианти могат да се прилагат и в геометрични задачи.

**Задача 11.** Изпъкнал $n$-ъгълник е разрязан на триъгълници с помощта на няколко свои диагонала, непресичащи се във вътрешни точки. Докажете, че броят на триъгълниците не зависи от начина на разрязване.

**Решение:** Сборът от ъглите на всеки триъгълник е 180°, а сборът от ъглите на $n$-ъгълника е $(n-2)\,.\,180°$. Следователно броят на триъгълниците е $n-2$ независимо от начина на разрязване на $n$-ъгълника.

**Задачи за полуинварианти**

Най-често полуинвариантите се използват, за да се докаже, че даден процес ще завърши; тоест че някаква операция не може да се извършва безкрайно. Примери са задачите за алгоритми по-горе, както и следната задача за VI клас от конкурса "Издирване на таланти" на сп. "Математика плюс", бр. 2/1996 г.

**Задача 1.** По окръжност са написани естествени числа. Между всеки две съседни числа записваме техния най-голям общ делител (като правим това едновременно по цялата окръжност), след което изтриваме старите числа. Докажете, че ще настъпи миг, когато всички числа ще бъдат равни.

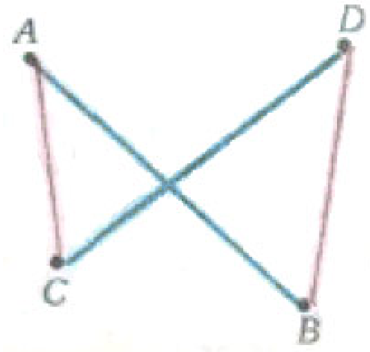
**Решение:** Полуинвариант е сборът на числата: той намалява винаги, когато има различни числа, защото НОД е по-малък или равен на по-малкото число. Понеже сборът на естествени числа е също естествено число, а не съществува безкрайна строго намаляваща редица от естествени числа, то в някой миг сборът ще спре да намалява. Това е възможно само когато числата са равни.

В тази задача, както и при алгоритмите, операцията беше дадена наготово. Обаче полуинвариантите могат да се използват и по друг начин — чрез тях може да се доказва съществуване на структури с някакви желани свойства. Обикновено започваме от структура, която не изпълнява някои изисквания, после я подобряваме стъпка по стъпка. Полуинвариантът показва колко далеч е текущата структура от желаната; той трябва да намалява строго. В тези задачи се налага сами да открием операция, подобряваща структурата.

Задачите до края на раздела са известни, има ги в множество източници: в статията "Этюды о полуинварианте" от Л. Д. Курляндчик и Д. В. Фомин, публикувана в руското списание "Квант", както и в статията "Полуинварианти" от Иван Симеонов, публикувана в бр. 2/1995 г. на сп. "Математика плюс".

**Задача 2.** Докажете, че $2n$ точки в равнината могат да бъдат свързани с $n$ отсечки, които две по две не се пресичат и нямат общи краища.

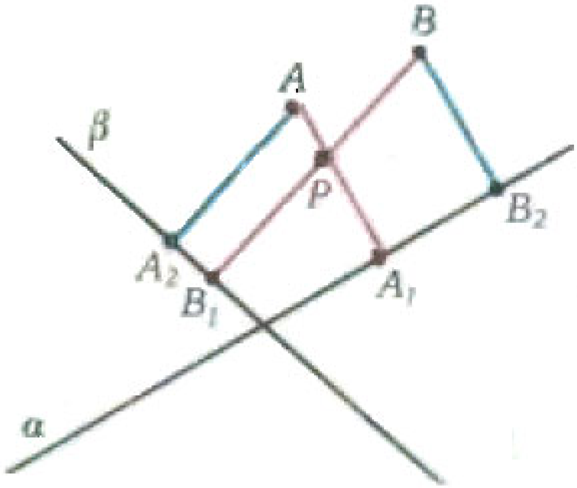
**Решение:** Отначало свързваме точките с $n$ отсечки без общи краища. Ако две отсечки, например $AB$ и $CD$, се пресичат, заменяме ги с непресичащите се $AC$ и $BD$.



Полуинвариант на тази операция е сборът от дължините на отсечките: той намалява при замяна на отсечки (следва от неравенството на триъгълника). Но възможните групирания на точките по двойки са краен брой, следователно сборът от дължините на отсечките не може да намалява безкрайно. Все някога ще се стигне до положение, при което сборът не може да намалява повече. Тогава операцията ще бъде неприложима, тоест няма да има пресичания.

**Задача 3.** В равнината са дадени $n$ прави, сред които няма успоредни, и $n$ точки. Да се докаже, че от дадените $n$ точки можем да спуснем общо $n$ перпендикуляра — по един към всяка права и по един от всяка точка, и то така, че перпендикулярите да не се пресичат два по два.

**Решение:** Започваме с $n$ перпендикуляра — по един към всяка права и по един от всяка точка. Ако те два по два не се пресичат, задачата е решена. Иначе нека перпендикулярите $AA_1$ и $BB_1$ от $A$ и $B$ към $\alpha$ и $\beta$ съответно се пресичат в някаква точка $P$. Заменяме ги с перпендикулярите $AA_2$ и $BB_2$ от $A$ и $B$ към $\beta$ и $\alpha$ съответно.




Тази замяна намалява сбора от дължините на перпендикулярите. Наистина, тъй като перпендикулярът е най-късото разстояние от точка до права, то следва, че са изпълнени неравенствата

$$AP + PB_1 \geq AA_2$$

и

$$BP + PA_1 \geq BB_2$$

и поне едно от тях е строго. Като ги съберем, получаваме строго неравенство:

$$(AP + PB_1) + (BP + PA_1) > AA_2 + BB_2 .$$

Прегрупираме събираемите в лявата страна:

$$(AP + PA_1) + (BP + PB_1) > AA_2 + BB_2 ,$$

тоест

$$AA_1 + BB_1 > AA_2 + BB_2 ,$$

което искахме да докажем.

Тъй като възможните групирания между точките и правите са краен брой, редицата от размени не може да бъде безкрайна. Рано или късно ще получим $n$ перпендикуляра, чиято обща дължина не може да бъде намалена. Това значи, че операцията е неприложима, тоест никои два перпендикуляра не се пресичат.

**Задача 4.** Всеки народен представител има не повече от трима врагове в парламента. Докажете, че парламентът може да се раздели на две палати така, че всеки народен представител да има не повече от един враг в своята палата.

**Решение:** Разделяме произволно парламента на две (непразни) палати Ако депутат има поне двама врагове в своята палата, преместваме го в другата. Така възникват нула или една двойки врагове в новата му палата, а изчезват две или три двойки врагове в старата му палата. Следователно тази операция намалява общия брой на двойките врагове. Но той е цяло неотрицателно число, затова не може да намалява безкрайно. Някога ще стигнем до разпределение, за което броят на двойките врагове не може да намалява повече. Следователно операцията е станала неприложима, което значи, че в този миг всеки депутат има не повече от един враг в своята палата.

**Задачи от състезания**

**Задача 1** (за VIII клас от XIX Всесъюзна олимпиада по математика в СССР, април 1985 г.). Целите числа от 1 до $2n$ включително са разделени на две групи, по $n$ числа в група. Нека $a_1 < a_2 < \ldots < a_n$ са числата от първата група, номерирани в нарастващ ред, $b_1 > b_2 > \ldots > b_n$ са числата от втората група, номерирани в намаляващ ред. Да се докаже, че

$$\left| a_1 - b_1 \right| + \left| a_2 - b_2 \right| + \ldots + \left| a_n - b_n \right| = n^2.$$

**Решение:** Ще докажем, че лявата страна е инвариант относно операцията размяна на две съседни стойности. Нека $a$-тата и $b$-тата са наредени заедно в нарастващ ред, например

$$a_1 < a_2 < b_n < a_3 < b_{n-1} < \ldots < a_k < b_p < \ldots < a_n < b_1.$$

Какво ще стане, ако разменим две съседни числа — едно $a$ и едно $b$, например $a_k$ и $b_p$ ? Разменяме не самите числа в редицата, а само стойностите на съответните променливи; тоест след размяната

$$a_1 < a_2 < b_n < a_3 < b_{n-1} < \ldots < b_p < a_k < \ldots < a_n < b_1.$$

Тъй като са съседни стойности, те се различават с единица.

Първи случай: Нека $a_k < b_p$ преди размяната ($a_k > b_p$ след размяната) и $k < p$. Следователно $a_k$ се е увеличило с единица и се е приближило към $b_k$, затова събираемото $\left| a_k - b_k \right|$ е намаляло с единица. Освен това $b_p$ също е намаляло с единица и се е отдалечило от $a_p$, затова $\left| a_p - b_p \right|$ се е увеличило с единица. И така, променили са се само две събираеми: едното е намаляло с единица, другото се е увеличило с единица. Затова сборът запазва стойността си.

Втори случай: Нека $a_k < b_p$ преди размяната ($a_k > b_p$ след размяната) и $k > p$. Следователно $a_k$ се е увеличило с единица и се е отдалечило от $b_k$, поради което събираемото $|a_k - b_k|$ се е увеличило с единица. А пък $b_p$ е намаляло с единица и се е приближило към $a_p$, затова $|a_p - b_p|$ е намаляло с единица. Щом като едно събираемо е намаляло с единица, а друго се е увеличило с единица, то сборът е запазил стойността си.

Трети случай: Нека $a_k < b_p$ преди размяната ($a_k > b_p$ след размяната) и $k = p$. Сега се е променило само едно събираемо: $|a_k - b_k|$; обаче се е променил само знакът на разликата; стойността на събираемото се е запазила (заради модула). Следователно и сборът е запазил своята стойност.

Аналогично се разглеждат случаите, когато $b_p < a_k$ преди размяната. (Можем да не ги разглеждаме, защото ще разместваме числата така, че всички $a$-та да отидат надясно от всички $b$-та.)

И тъй, стойността на сбора е инвариант относно разглежданата операция — размяна на групите на две числа, чиито стойности се различават с единица. С нейна помощ можем да докараме всички $a$-та вдясно от всички $b$-та: $a_k = n + k$ (тоест $a$-тата са числата $n + 1 < n + 2 < n + 3 < \ldots < 2n$ в този ред), $b_k = n + 1 - \mathrm{k}$ (тоест $b$-тата са числата $n > n - 1 > n - 2 > \ldots > 1$ в този ред),

$$\sum_{k=1}^{n} |a_k - b_k| = \sum_{k=1}^{n} (a_k - b_k) = \sum_{k=1}^{n} (2k - 1) = 1 + 3 + 5 + \ldots + (2n - 1) = n^2.$$

Понеже сборът е инвариант, то той е равен на $n^2$ и за всяко друго разпределение (от всяко разпределение можем да стигнем до разгледания частен случай).

**Задача 2** (дадена на XLIX Московска математическа олимпиада, 1986 г.). Квадратно поле е разделено на сто еднакви квадратни участъка, разположени в десет реда и десет стълба. В девет от участъците са пораснали бурени. Участък, заразен с бурени, остава буренясал и по-нататък. Участък, в който сега няма бурени, догодина ще буреняса, ако поне два от съседните му участъци имат бурени сега. (Два участъка се смятат за съседни, ако имат обща страна.) Да се докаже, че полето никога няма да буреняса изцяло.

**Решение:** Като се разгледат възможните случаи за нов заразен участък (с два, три и четири съседни участъка, буренясали по-рано), се установява, че обиколката на буренясалата част не нараства, т.е. всяка година намалява или остава същата. (Под “обиколка” разбираме сбора от дължините на отсечките, които са вътрешни за големия квадрат и отделят буренясал от чист участък, плюс отсечките от контура на големия квадрат, ограждащи буренясал участък.) Отначало обиколката на буренясалата част не надхвърля $4 \,.\, 9 = 36 < 40$. Щом не расте, тя винаги ще остане по-малка от 40. За да буреняса цялото поле, разглежданата обиколка трябва да стане 40. Понеже тя не може да стане 40, то не е възможно полето да буреняса изцяло.

**Задача 3** (из материалите на международното жури от МОМ 1994; предложение от Швеция). Около кръгла маса са насядали 1994 момичета, които играят с колода от $n$ карти. Отначало цялата колода е у едно момиче. Всеки ход на играта протича така: ако поне едно момиче има поне две карти, едно от тези момичета (произволно избрано) трябва да даде по една карта на всяка от двете си съседки. Играта свършва тогава, когато всяко момиче има най-много една карта. Да се докаже, че играта не може да свърши при $n \geq 1994$.

**Решение:** Играта не свършва при $n > 1994$ поради принципа на Дирихле: у поне едно момиче ще има най-малко две карти, както и да се разпределят.

Интересният случай е $n = 1994$. Той се решава с подходящ инвариант — четността на общия брой карти у момичетата, седящи на нечетните места (№ 1, № 3, № 5, ... , № 1993), като № 1 е мястото на момичето, което отначало държи всички карти. Общият брой карти у тези момичета отначало е равен на $1994 + 0 + 0 + ... + 0 = 1994$, а на всеки ход расте или намалява с 2, следователно винаги е четно число. При $n = 1994$ играта би могла да завърши само когато всяко момиче има точно една карта. Тогава общият брой карти у момичетата на нечетни места би бил $1994 : 2 = 997$, т.е. нечетно число, което е невъзможно. Следователно при $n = 1994$ играта не може да завърши.

**Задача 4** (задача 6 от контролните за определяне на българския отбор за Международната олимпиада по математика за средношколци през 1994 г.). Страните на правилен 99-ъгълник са оцветени с три цвята по следния начин: червена, синя, червена, синя, . . . , червена, синя, жълта (обходени по часовника). Разрешено е да преоцветим произволно избрана страна, стига новият ѝ цвят да е различен от цветовете на всяка от двете ѝ съседни страни. Възможно ли е след няколко преоцветявания да се получи следната редица от цветове: червена, синя, червена, синя, . . . , червена, жълта, синя (обходени по часовника)? Тази редица може да започва от друга страна на 99-ъгълника, но непременно трябва да го обикаля в същата посока (по часовника).

**Решение:** Не може да се постигне такова оцветяване. Да заменим цветовете с числа: червено = 1, синьо = 2, жълто = 3. Тези числа, отчитани по часовника от произволно избрано място, да означим с $x_1, x_2, x_3, \ldots, x_{99}$ и да разгледаме следния израз: $A = (x_1 - x_2)(x_2 - x_3)(x_3 - x_4) \ldots (x_{98} - x_{99})(x_{99} - x_1)$. Този израз не зависи от избраното начално място: промяната на началното място води само до разместване на множителите, при което произведението не се променя.

Промяна на цвета на някоя страна е възможна само в следната ситуация: подредицата $(x\,;\ y\,;\ x)$ се заменя с $(x\,;\ z\,;\ x)$, където $\{\, x\,; y\,; z \,\} = \{\, 0\,; 1\,; 2 \,\}$, тоест числата $x$, $y$ и $z$ са две по две различни. Това води до замяната на два поредни множителя $(x - y)(y - x)$ в $A$ с други два множителя: $(x - z)(z - x)$. Ето защо стойността на израза $A$ не е инвариант, обаче знакът му е инвариант: $(x - y)(y - x) = -(x - y)^2 < 0$ и $(x - z)(z - x) = -(x - z)^2 < 0$, тоест заменяме отрицателно произведение с отрицателно произведение.

Отначало $A = [(-1)(+1)(-1)(+1)] \dots [(-1)(+1)(-1)(+1)][(-1)(-1)(+2)] = 2 > 0$ (броят на чифтовете квадратни скоби с четири множителя е двайсет и четири). Следователно стойността на израза $A$ винаги остава положителна.

Желаното крайно оцветяване би довело израза $A$ до отрицателна стойност: $A = [(-1)(+1)(-1)(+1)] \dots [(-1)(+1)(-1)(+1)][(-2)(+1)(+1)] = -2 < 0$. Както знаем, изразът $A$ приема само положителни стойности. Затова желаното оцветяване не може да бъде постигнато чрез разрешените ходове.

**Задача 5** (XL Национална олимпиада по математика за средношколци, задача 6 от IV кръг — София, юни 1991 г.). Шахматна дъска $n \times n$ ($n \geq 2$) се запълва с бели и черни пулове. На произволно поле се поставя черен пул. На всеки следващ ход се поставя бял пул на някое празно поле и се променят цветовете на съседните пулове (две полета са съседни, ако имат обща страна). Това продължава, докато цялата дъска се запълни с пулове. Да се докаже, че накрая ще има поне един черен пул.

**Решение:** Да свържем с отсечки центровете на всеки две съседни полета. Отсечките са четен брой: колкото са водоравните, толкова са и отвесните, защото дъската е квадратна. Ще изтриваме отсечка в мига, когато слагаме втори пул в краищата ѝ.

Инвариант: Броят на неизтритите отсечки плюс броя на черните пулове е нечетно число.

Доказателство: с математическа индукция по номера на хода.

База: След първия ход са налице всички отсечки (чийто брой е четно число) и има един черен пул, а сборът на четно число с единица е нечетно число.

Индуктивна стъпка: Да предположим, че инвариантът важи след някой ход. На следващия ход добавяме бял пул (което не променя броя на черните пулове, нито броя на оставащите отсечки, следователно не променя четността на сбора от двете бройки), извършваме $x$ (тоест 0, 1, 2, 3 или 4) промени на цветове на пуловете на съседни заети полета и изтриваме $x$ отсечки (свързващи текущото поле със заетите $x$ съседни полета). Всяка промяна на цвят на пул променя броя на черните пулове с единица, следователно променя четността на броя им. Всяко изтриване на отсечка намалява броя на отсечките с единица, затова променя четността на техния брой. Четността на сбора от двете бройки се променя с четността на $2x$, което е четно число. Тоест въпросният сбор не променя четността си при поредния ход. По предположение този сбор е бил нечетен преди хода, следователно остава нечетен след хода.

Инвариантът е доказан. С негова помощ довършваме решението на задачата. В крайното положение на дъската всички отсечки са вече изтрити, тоест броят на неизтритите отсечки е 0 (четно число). Щом сборът е нечетен, то другото събираемо, т.е. броят на черните пулове, е нечетно число. Затова то не е 0. Следователно има поне един черен пул на дъската в нейното крайно положение.