Даниел Велинов
Скопје

ИНВАРИЈАНТИ И МОНОВАРИЈАНТИ

Концептот на инваријанти и моноваријанти е важен концепт и многу често се користи во решавање на покомплицирани проблеми. Моноваријанта е бројна карактеристика која се менува монотоно (таа е растечка или опаѓачка), додека инваријанта е бројна карактеристика која не се менува. Овие концепти се особено важни кога работиме со проблеми од комбинаторика и можат да бидат од голема помош при конструирање на алгоритми. Моноваријантите најчесто даваат одговор на прашањето: Што да правиме понатаму?

Обично проблемите со инваријанти се прилично лесни кога ќе биде најдена вистинската инваријанта (моноваријанта), но нејзиното наоѓање некогаш може да биде навистина тешко. Во суштина ноѓањето на вистинската инваријанта е уметност и тоа е она нешто што овие проблеми ги прави тешки. Не постои некое прецизно правило за наоѓање на инваријанта во даден проблем, но при решавањето на проблеми може да се размисли за следниве работи:

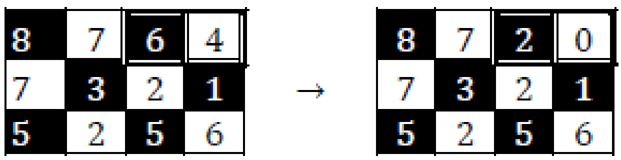
- При боење: Обои ги сите полиња во шемата со две или повеќе бои. Обично боењето како шаховска табла е добар избор, но некогаш и останатите боења можат да бидат од помош. Разгледувај квадрати од сите бои посебно.
- Алгебарски изрази: За дадено множество на вредности, разгледај ги нивните разлики, нивните збирови, збировите на квадратите или нивниот производ. Ако работиш со цели броеви, обиди се да гледаш по модул n.
- Темиња и ребра: За проблемите со графови разгледувај ги сите можни форми. Колку гранични ребра имаат тие? Колку темиња?
- Инверзии: Ако има пермутација на некои броеви, разгледувај го бројот на инверзии, односно бројот на парови (i, j) такви што i и j се запишани во обратен редослед од почетниот. Апсолутниот број на инверзии и нивната парност многу често се инваријанти.
- Цели броеви и рационални броеви: Може ли да се најде позитивен цел број кој постојано опаѓа? Дали именителот на дропката постојано опаѓа?
- Симетрии: Може ли во секој чекор, да има фигура која е симетрична на некој начин? Можеби логички може да се споруваат објекти и тие така спарени објекти се секогаш во исто множество? Можеби проблемот може да се подели во два идентични потпроблеми? Ова размислување е особено е корисно во теорија на игри.

Инваријантите може да се применат на многу интересни начини, но секогаш е различно од ситуација до ситуација. Горенаведените ситуации можат само да насочат како би размилувале во некоја конкретна ситуација, но тоа не гарантира дека ќе успееме да го решиме проблемот. Со решавање на повеќе проблеми со инваријанти и моноваријанти, нивното утврдување би било полесно.

Во следните неколку примери ќе илустрираме како инваријантите и моноваријантите играат многу важна улога во конструирањето на алгоритми и нивна проверка дека тие навистина функционираат.

Пример 1. (IMO shortlist, 1989) Во секое од полињата на шаховска табла $m\times n$ е напишан по еден природен број. Дозволен потег е да се додаде број цел број k на секои два соседни броеви така што се добиваат ненегативни броеви (соседни броеви се броеви кои лежат во соседни полиња, додека соседни полиња се оние кои имаат заедничка страна). Најди потребен и доволен услов за сите броеви во полињата бидат нула, после конечно многу потези.

Решение. Да забележиме дека во еден потег го додаваме истио број на две полиња, од кои едното е бело поле и едното е црно поле, бидејќи станува збор за шаховска табла. Ако со S_b и S_w ги означиме збировите на броевите запишани во црните, односно белите полиња. Тогаш $S_b - S_w$ е инваријанта. Тоа значи дека ако сите броеви се нула на крај, тогаш $S_b - S_w = 0$ на крајот, па $S_b - S_w = 0$ на почетокот. Ова значи дека овој услов е потребен. Сега ќе покажеме дека всушност овој услов е и доволен.



Да претпоставиме дека a,b,c се броевите запишани во полињата A,B,C, соодветно каде A,B,C се полиња такви што A и C се соседни полиња на B. Ако $a\le b$, можеме да додадеме на двете полиња $(-a)$, па добиваме на местото на a 0. Ако $a\ge b$, тогаш додаваме $(a-b)$ на b и c. Тогаш b станува a и сега можеме на двете полиња да додадеме $(-a)$ правејќи ги и двете нула. Па на овој начин имаме алгоритам за трансформирање на природен број во 0. Применувајќи го ова во секој ред, можеме да ги направиме сите нули освен последните две полиња. Сега сите колони се нули освен последните две. Сега ако го примениме овој алгоритам почнувајќи од врвот на овие две колони, додека овие две соседни ненулти броеви не останат. Последните два броја мора да бидат еднакви бидејќи $S_b = S_w$. Па, може целата шаховска табла да се запише со конечен број на потези целата со нули.

Пример 2. (New Zealand IMO TST, 2011) Нека $2n$ луѓе седат на кружна маса и нека на нив им се поделени m колачиња. Колачињата може да се подаваат под следниве услови:

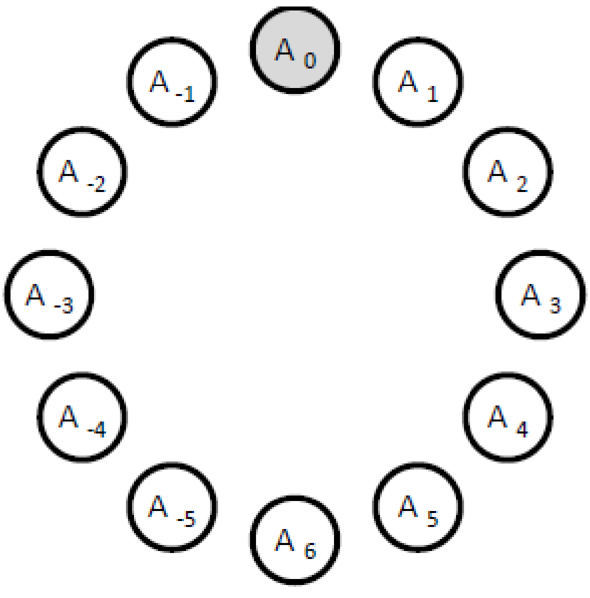
а) Секој може да подаде колаче само на соседите б) Ако некој подаде на некого колаче, тогаш тој мора и да изеде едно колаче.

Нека A е еден од тие луѓе. Најди го најмалиот m така што без разлика колку колачиња се поделени првично, постои стратегија за подавање на колачињата така што A да добие најмалку едно колаче.

Решение. Нека ги означиме луѓето со $A_{-n+1}, A_{-n+2}, ..., A_0, A_1, A_2, ..., A_n$, така што $A = A_0$. Исто така означуваме $A_{-n} = A_n$. Припишуваме тежина $\frac{1}{2^{|i|}}$ за секое колаче кое го поседуваме лицето A_i. Ако на пример A_3 му додаде колаче на A_2, тогаш тежината на колачето расте од $\frac{1}{8}$ до $\frac{1}{4}$. Да збележиме дека и A_3 мора да изеде едно колаче (колаче со тежина $\frac{1}{8}$) во овој чекор. Од ова можеме да заклучиме дека тежината на колачињата се нема променето. Уште попрецизно, ако A_i има a_i колачиња за секој i, па тогаш вкупната тежина на колачињата е

$$W = \sum_{i=-n+1}^{n} \frac{a_i}{2^{|i|}}.$$

Кога и да се случува едно колаче да биде подадено на A_0 (од $A_{\pm i}$ на $A_{\pm(i-1)}$ за i позитивно) едно колаче треба да биде изедено и друго колаче ја зголемува својата тежина за двапати, па вкупната тежина се намалува. Па, вкупната тежина е моноваријанта.



Ако $m < 2^n$, тогаш ако сите колачиња почетно се доделени на A_n, почетната вкупна масе е $\frac{m}{2^n} < 1$. Па така вкупната маса е секогаш помала од 1 (бидејќи никогаш не расте со нашите операции), па A_0 не може да добие колаче (ако A_0 добие колаче, тогаш тоа би имало тежина 1). Следува мора $m \geq 2^n$.

Сега ќе покажеме дека за $m \geq 2^n$, можеме секогаш да обезбедиме дека A_0 ќе добие колаче. Интуитивно ќе ја следиме следнава идеја: Алгоритмот кој требада го најдеме никогаш не треба да додава колаче од A_0 на некој друг, бидејќи во спротивно ќе ја намалиме нашата моноваријанта. Тоа значи дека во секој чекор ќе треба да додаваме колачиња кон A_0.

Сега треба да најдеме начин накој начин A_n треба да додава колачиња, бидејќи и девете насоки на додавање со кон A_0 (ова бидејќи A_0 и A_n се дијаметрално спротивни).Ова не води кон тоа да разгледуваме нова големина со цел да ги разликуваме двете насоки во кои A_n може да подава колачиња. Нека W_+ е збирот на масите на колачињата кои ги имаат $A_0, A_1, A_2, ..., A_n$ и нека W_- е збирот на масите на колачињата кои ги имаат $A_0, A_{-1}, A_{-2}, ..., A_{-n}$. Без губење на општоста може да претпоставиме дека $W_+ \geq W_-$. Ова сугерира дека мора A_n да додава колачиња само на A_{n-1} и ние треба да работиме само во полукругот кој ги содржи само ненегативните индекси, бидејќи овој полукруг има поголема тежина. Со алгоритмот правиме A_n да додаде што е можно повеќе колачиња на A_{n-1}, па потоа A_{n-1} додава што е можно повеќе колачиња на A_{n-2} и така натаму се додека A_0 не добие колаче. Но, ова важи ако и само ако $W_+ \geq 1$. Јасно е дека $W_+ \geq 1$ е потребен услов бидејќи W_+ е моноваријанта во нашиот алгоритам, па останува уште да покажеме дека е доволен услов.

Нека $W_+ \geq 1$. Да забележиме дека нашиот алгоритам го остава W_+ инваријанта. Да претпоставиме дека нашиот алгоритам завршува, односно не можеме да подадеме повеќе колачиња од било кое A_i, каде i е позитивен, понатаму и A_0 нема колачиња. Тогаш $A_1, A_2, ..., A_n$ имаат најмногу едно колаче на крајот (ако има некој повеќе од едно, тогаш тој може да подаде едно и да изеде едно, па нашиот алгоритам не би бил завршен). Тогаш $W_+ \leq \frac{1}{2} + \frac{1}{4} + ... + \frac{1}{2^n} < 1$, што е контрадикција со фактот дека W_+ е инваријанта поголема од 1. Значи добивме дека $W_+ \geq 1$ е доволен услов за да нашиот алгоритам да фунцкионира.

Конечно, докажавме дека мора $W_+ \geq 1$, при претпоставка $W_+ \geq W_-$. Сега да забележиме дека секое колаче влијае со најмалку $\frac{1}{2^{n-1}}$ на збирот $(W_+ + W_-)$, бидејќи секое клаче има маса најмалку $\frac{1}{2^{n-1}}$, освен колачињата кај A_n. Колачињата кај A_n се броени двапати бидејќи тие придонесуваат и на W_+ и W_-, па тие исто така придонесуваат со $\frac{1}{2^{n-1}}$ на збирот. Бидејќи имаме најмалку 2^n колачиња, $W_+ + W_- \geq 2$, па $W_+ \geq 1$, со штопокажавме дека алгоритамот навистина функционира.

Пример 3. (IMO shortlist, 1994) Петар има 3 сметки во банка, секоја од нив со цел број на долари на неа. Нему му е дозволено да префрла пари од една сметка на друга сметка така што парите на втората сметка се дуплирани. Докажи дека Петар секогаш може да ги префрли сите пари на само две сметки. Дали секогаш може да ги пренесе сите пари на само една сметка?

Решение. Прашањето дали Петар може да ги префрли сите пари на една сметка е тривијално: Ако вкупната сума на долари кои ги има Петар на сите три сметки е непарна, тогаш не секогаш е можно да се префрлат сите пари на една сметка.

Сега да го докажеме првиот дел од задачата. Нека A, B, C, каде $A \leq B \leq C$, е бројот на долари на првата, втората и третата сметка во одреден момент. Ако $A = 0$ на почетокот, нема што да се докажува. Затоа, $A > 0$. Ако користиме некој алгоритам вредностите на A, B, C ќе се менуваат. Нашата цел со овој алгоритам би била монотоно строго да направиме да опаѓа вредноста на $\min\{A, B, C\}$, што ќе обезбеди дека во даден момент ќе завршиме со $\min\{A, B, C\} = 0$ со што ќе биде докажано тврдењето. Многу едноставен и корисен алгоритам кој монотоно го намалува е Евклидовиот алгоритам. Нека $B = qA + r$, каде $0 \leq r < A$. Наша цел е сега до го намалиме бројот на долари од втората сметка од B до r. Бидејќи $r < A$, бројот $\min\{A, B, C\}$ ќе биде намален што е наша цел.

Бидејќи бидејќи задачата вклучува дуплирање на дадени броеви, добра идеја е да се реазгледува бинарна репрезентација на броевите. Нека $q = m_0 + 2m_1 + 2^2 m_2 + ... + 2^k m_k$ е бинарна репрезентација на q, каде $m_i = 0$ или $m_i = 1$, за секое i. За да го намалиме B до r во нашиот алгоритам, ние префрламе пари на првата сметка. Префрлањето е од втората сметка ако $m_{i-1} = 1$ и од третата сметка ако $m_{i-1} = 0$. Бројот на долари на првата сметка почнува со A и се дуплира во секој чекор со горниот алгоритам. Па, завршуваме на првата сметка со $A(m_0 + 2m_1 + ... + 2^k m_k) = Aq$ долари од втората сметка на првата сметка, па на втората сметка има останато $B - Aq = r$ долари. Успеавме да го намалиме $\min\{A, B, C\}$, со што е најден алгоритам кој фунцкионира, па првиот дел од задачата е докажан.

Понекогаш додека работиме со алгоритми кои монотоно намалуваат одредена вредност (монотоно ја зголемуваат), во одредени ситуации нашиот алгоритам може да не работи. Многу често овие потечкотии ги надминуваме на слендиот начин: Велиме дека една позиција е добра ако можеме да ја намалуваме моноваријантата со нашиот алгоритам. Во спротивно велиме дека оваа позиција е лоша. Сега креираме алгоритам кој лошите позиции ги претвора во добри позиции, без да ја зголеми моноваријантата. Го користиме првиот алгоритам кога има услови за тоа, а кога имаме проблем да го користиме, тогаш го користиме вториот алгоритам се додека не дојдеме до добра позиција, кога пак почнуваме да го користиме првиот алгоритам. Следниот малку покомплициран пример ја илустрира оваа идеја.

Пример 4. (USAMO, 2003 Problem 6) Во темињата на правилен шестоаголник се напишани шест природни броеви чиј збир е 2003. Берт може да ги прави следните потези: Тој може да избере едно теме и да го замени бројот запишан во него со апсолутната вредност на разликата помеѓу броевите запишани во соседните темиња. Докажи дека Берт може после конечен број на потези да добие нули во сите шест темиња.

Решение. Алгоритмот кој ќе биде конструиран користи дека 2003 е непарен број. Нека збирот на позиција е зборот од броевите запишани во шестте темиња и нека максимумот го означува максималната вредност од шестте броеви запишани во темињата на шестаголникот. Нека A, B, C, D, E, F се броевите запишани во темињата во тој редослед. Наша цел ќе биде монотоно да го намалуваме максимумот. Да забележиме дека максимумот никогаш не може да расте во оваа ситуација.

Потребни ни се два поталгоритми:

а) Создавање на “добра позиција”: Од позиција со непарен збир, одиме на позиција со точно еден непарен број;

б) Намалување на моноваријантноста: Од позиција со точно еден непарен број одиме на позиција со непарна сума и строго помал максимум или одиме на сите нули во темињата.

За а), бидејќи $(A+B+C+D+E+F)$ се непарни, без губење на општоста можеме да претпоставиме дека $(A+C+E)$ е непарен. Ако точно еден од A, C, E е непарен, нека претпоставиме дека A е непарен. Тогаш ја правиме следната низа од потези: B, F, D, A, F (овде потегот е означен со темето на кое што се прави потегот). На овој начин ние завршуваме со ситуација во која само B е непарен и останатите темиња стануваат парни, а со тоа сме готови со а). Другата можност е A, C, E да бидат непарни. Во овој случај ја правиме низата од потези (B, D, F, C, E). После овие потези само A е непарен.

Сега сме спремни да го примениме б), чекорот кој всушност на намалува нашата моноваријанта. Сега, во овоа ситуација имаме едно теме во кое е запишан непарен број и нека тоа е темето A. Повторно имаме две ситуации. Ако максимумот е парен, тогаш тоа е еден од броевите B, C, D, E или F. Сега правиме потези на B, C, D, E и F во следниот редослед: Ако максимумот е непарен, тогаш тоа е A. Ако $C=E=0$, тогаш низата од потези (B,F,D,A,B,F) ни овозможува да добиеме нули во сите темиња. Во спротивно нека претпоставиме дека едно од C и E не е нула, па нека $C>0$ (случајот кога $E>0$ е сличен). Во овој случај ги правиме потезите (B,F,A,F). Лесно се проверува дека со овие потези се намалува максимумот и добиваме непрен збир.

Па, почнувајќи со непарен збир, го применуваме а) ако е потребно, после што го применуваме б). Ова го намалува максимумот и добиваме непарен збир (или на крај не остава со сите нули со што сме готови со доказот), па целата постапка може да се повторува се додека максимумот не стане 0.

Пример 5. (APMO 1997, Problem 5) n луѓе се седната на кружна маса. На нив им се поделени вкупно nk монети, при што не мора да добијат сите ист број на монети. Еден потег е да се предаде една монета помеѓу два соседи. Да се најде алгоритам за најмалку можни потези со кој сите луѓе седнати на кружната маса ќе имаат ист број на монети.

Решение. Наша цел е секој да заврши со k монети. Нека луѓето ги означиме со $1,2,3,...,n$ по ред (n е сосед со 1 бидејќи тие седат на кружна маса). Нека претпоставиме дека i има c_i монети. Ја воведуваме променливата $d_i=c_i-k$, која покажува колку i е блиску до тоа да го има посакуваниот број на монети. Да го разгледаме следниот број

$$X=|d_1|+|d_1+d_2|+|d_1+d_2+d_3|+...+|d_1+d_2+...+d_{n-1}|.$$

Јасно $X=0$ ако и само ако сите имаат по k монети, па наша цел е да направиме $X=0$.

Причината за овој избор на X е што при предавањето на една монета помеѓу j и $j+1$, за $1\le j\le n-1$ го менува X точно за 1, бидејќи само делот

$$|d_1+d_2+...+d_j|$$

се менува за 1.

Па, X е моноваријанта и прилично едноставно е да се контролира (освен кога се предава монета од 1 на n и обратно). Нека $s_j=d_1+d_2+...+d_j$.

Тврдиме дека се додека $X>0$ можно е да го намалиме X за 1 со потег предавање на монета од j на $j+1$ за некое $1\le j\le n-1$. Го користиме следниов алгоритам. Без губење на општоста можеме да претпоставим дека $d_1\ge 1$. Го земаме првиот j таков што $d_{j+1}<0$. Ако $s_j>0$, тогаш само правиме префрлање од j на $j+1$. Ова го намалува X за 1, бидејќи го намалува $|s_j|$ за еден. Другата можност е $s_j=0$, што значи дека $d_1=d_2=...=d_j=0$ (d_{j+1} е првиот негативен број). Во овој случај, земаме прво $m>i+1$ така да $d_m\ge 0$. Тогаш $d_{m-1}<0$ по претпоставката за m, па имаме предавање на монета од m на $(m-1)$. Да забележиме дека сите броеви пред d_m се или 0 или помали од 0 и $d_{m-1}<0$, па s_{m-1} е помало од 0. Нашиот потег го зголеми s_{m-1} за еден и со тоа е намалено $|s_{m-1}|$ за еден, па X е намалено за еден.

Имаме дека во било која ситуација ние можеме да го намалиме X за еден со предеавање на една монета помеѓу j и $j+1$, за некој $1 \le j \le n-1$. Овде сеуште не го разгледавме ефектот на предавање на монета помеѓу 1 и n и обратно. Всушност нашиот алгоритам би бил следниов: Ако со било кој потег го намалиме X со предавање на монета од 1 на n и обратно, тогаш го правиме тоа. Во секој друг случај го користиме алгоритамот опишан погоре за да го намалиме X за еден.

Користена литература

1. P. A. Sriram, Olympiad Combinatorics, 2014.
2. M. Rognlie, Invariants, Monovariants and Extrema, Duke University, 2009.
3. M. Lavrov, Invariants, ARML Practice, 2012.