Информации, Ефикасност и Рекурзии

Велинов Даниел, Градежен факултет-СКОПЈЕ

Во следните неколку проблеми ќе илустрираме како се собираат информации и извршуваат операции со нив ефикасно, односно операциите кои ќе бидат извршени со нив ќе бидат со минимален број на чекори. Определувањето на некоја информација или доаѓањето до неа со најмал можен број на чекори или прашања е многу важно во информатичките технологии.

Следниот пример е прост и многу добро познат во компјутерските науки.

Пример 1. (Сортирачки алгоритам) Нека се дадени n реални броеви кои сакаме да ги сортираме (да ги запишеме во неопаѓачки редослед) користејќи што е можно помалку споредби (во една споредба земаме два броја a и b и проверуваме дали $a<b$, $b<a$ или $a=b$). Јасно, нив можеме да ги сортираме ако ги направиме сите можни $\frac{n(n-1)}{2}$ споредби. Дали може тоа да се направи со помалку споредби?

Решение. Да. Користиме рекурзивен алгоритам. Нека $f(n)$ е бројот на споредби кој е потребен за множество од n броеви. Множеството од n броеви го делиме на две множества со големина $\frac{n}{2}$ (или ако n е непарен, со големина $\frac{n-1}{2}$ и $\frac{n+1}{2}$. Во остатокот од овој проблем, претпоставуваме дека n е парен заради поедноставна работа.) Сега ги сортираме овие две множества посебно. Ова бара $2f\left(\frac{n}{2}\right)$ споредби. Да претпоставиме дека тие се сортирани и дека важи $a_1 \le a_2 \le \ldots \le a_{\frac{n}{2}}$ и $b_1 \le b_2 \le \ldots \le b_{\frac{n}{2}}$. Наредно е што ќе направиме е да ги комбинираме овие две множества или да ги споиме. Прво ги споредуваме a_1 и b_1. Следува после споредба помеѓу a_i и b_j, ако $a_i \le b_j$, ги споредуваме a_{i+1} и b_j и ако $b_j < a_i$ ги споредуваме b_{j+1} и a_i. Оваа постапка завршува после најмногу n споредби, после кои ние комплетно ќе ја сортираме низата од n броеви. Користевме најмногу $2f\left(\frac{n}{2}\right)+n$ споредби, па $f(n) \le 2f\left(\frac{n}{2}\right)+n$.

Од оваа рекурзија, со помош на принципот на математичка индукција можеме да покажеме дека $f(2^k) \le k2^k$ и поопшто за n броеви, потребниот број на споредбие од ред $n\log_2 n$, што е поефикасно од тривијалната граница $\frac{n(n-1)}{2}$ која е од ред n^2.

Пример 2. (IMO shortlist, 1998) Карти кои се нумерирани од 1 до n се подредени случајно во ред при што $n \ge 5$. Во еден чекор, може да се избере било кој блок од последователни карти чии броеви се во растечки или опаѓачки редослед и да се заменат картите во тој блок. На пример, ако $n=9$, тогаш $91\underline{653}2748$ може да се промени во $91\underline{356}2748$. Докажи дека со најмногу $2n-6$ чекори, може картите да се наредат во растечки или опаѓачки редослед.

Решение. Ќе користиме рекурзивен алгоритам кој ситуацијата со n карти ја сведува на ситуација со $n-1$ карти. Нека $f(n)$ е минималниот број на чекори потребни произволен распоред на картите да се запише во растечки или опаѓачки редослед.

Да претпоставиме дека имаме пермутација која почнува со картата k. Во точно $f(n-1)$ потези, ние можеме да ги подредиме останатите $(n-1)$ карти и да ја добиеме низата $(k,1,2,...,k-1,k+1,...,n)$ или низата $(k,n,n-1,...,k+1,k-1,...2,1)$ и потоа со два потези да го добиваме бараното подредување. Значи, во било кој случај ни се потребни точно два дополнителни потези за да го добиеме бараното подредување, при што се искористени $f(n-1)+2$ потези, па $f(n)\le f(n-1)+2$.

Сега за да докажеме дека ова е точно за $n\ge 5$, доволно е да докажеме дека горната граница е точна за $n=5$ и потоа со индукција може да се покаже дека $f(n)\le f(n-1)+2$. За да докажеме дека $f(5)\le 4$, прво да забележиме дека $f(3)=1$ и $f(4)=3$. Лесно може да се покаже дека било која пермутација од 4 карти може да се запише во растечки или опаѓачки редослед со најмногу 3 потези. Сега нека е дадена произволна пермутација на $\{1,2,3,4,5\}$ и користиме еден потег да се осигураме дека или 1 или 5 се на некоја од крајните позиции. Сега останатите четири члена ги запишуваме во растечки или опаѓачки редослед со помош на претходниот случај кога 4 во најмногу 3 потези, така што сега целата низа е запишана во растечки или опаѓачки редослед. Па, најмногу 4 потези се потребни за 5 карти, со што сме готови со доказот.

Пример 3. (Russia, 2000) Тања избира природен број $X\le 100$ и Саша се обидува да го погоди тој број. Таа може да избере два природни броја M и N помали од 100 и да праша која е вредноста на $Н.З.Д(X+M,N)$. Докажи дека Саша може да го открие бројот кој Тања го избрала со најмногу седум прашања (броевите M и N можат да се менуваат со секое прашање).

Решение. Бидејќи $2^6<100<2^7$, претпоставуваме дека поопшто потребни се $\lceil \log_2 n\rceil$ погодувања, каде n е максималната можна вредност на X и $\lceil\cdot\rceil$ е таванска функција.

Нашата стратегија ќе биде да ги определиме цифрите на X во бинарен запис. Прво прашуваме за $Н.З.Д.(X+2,2)$. Ова ќе ни каже дали X е парен или непарен, па ќе ја знаеме цифрата на единиците во бинарниот запис на X. Ако X е парен, тогаш прашуваме за $Н.З.Д.(X+4,4)$. Ова ни кажува дали X е делив со 4 или не е делив. Во спротивно прашуваме за $Н.З.Д.(X+1,4)$. Ова ни кажува дали X е 1 или 3 по $\bmod 4$ (ако $Н.З.Д.(X+1,4)=4$, тогаш $X+1$ е деливо со 4, па $X\equiv 3 \pmod 4$). Со помош на оваа информација можеме да ја утврдиме наредната цифра од бинарниот запис на X. На пример, ако X е непарен и е 3 по (mod 4), последните две цифри на X во бинарен запис се 11. Нека сега претпоставиме дека $X \equiv i \pmod 4$. За да ја определиме наредната цифра прашуваме за $Н.З.Д.(X+(4-i),8)$. Тој може да биде или 4 или 8, според тоа дали $X \equiv i$ или $X \equiv 4+i \pmod 8$. Со ова ја добиваме наредната цифра. На пример, ако $X \equiv 3 \pmod 4$ и $X \equiv 7 \pmod 8$, тогаш последните три цифри на X во бинарен запис се 111, а ако $X \equiv 3 \pmod 8$, тогаш последните три цифри на X во бинарен запис се 011. Сега постапката е јасна како да ги определиме сите цифри во бинарен запис за бројот X. За ова се потребни k прашања, каде k е бројот на цифри во бинарниот запис на n (бидејќи $X \le n$, нема потреба од барање на дополнителни цифри во бинарниот запис на X), кое во најлош случај е еднакво на $\lceil \log_2 n \rceil$.

Пример 4. (Russia, 2004, Grade 9, Problem 3) На една маса има n кутии, каде n е парен и позитивен и во секоја кутија има по едно топче. Некои од топчињта се бели и бројор на бели топчиња е парен и поголем од 0. Во секој потег избираме де произволни кутии и прашуваме дали има најмалку едно бело топче во двете кутии (одговорот е да или не). Докажи дека после $(2n-3)$ прашања можеме да најдеме две кутии кои со сигурност содржат бели топчиња.

Решение. Да ги означиме кутиите од 1 до n. Прашуваме за паровите кутии $(1, j)$, каде $j = 2,3,...,n$. Ако во некој момент добиеме одговор не, тоа значи дека кутијата со реден број 1 не содржи бело топче. Тогаш за сите j за кои ќе добиеме одговор да за $(1, j)$, кутијата j содржи бело топче, со што сме готови. Единствена опција е ако добиеме одговор да за сите парови од кутии $(1, j)$, во кој случај има две опции: или кутијата 1 има бело топче или кутијата 1 нема бело топче и сите останати $(n-1)$ кутии имаат бели топчиња. Вториот случај отпаѓа бидејќи имаме парен број на кутии кои имаат бели топчиња, а $(n-1)$ е непарен број од условот на задачата. Следува кутијата 1 има бело топче. Сега прашуваме за паровите $(2, j)$, каде $j = 3,4,...,n$. Да забележиме дека имаме поставено вкупно $(n-1)+(n-2) = 2n-3$ прашања. Ако во некој случак добиеме одговор не, тогаш кутијата 2 нема бело топче и сите одговори да ни кажуваат каде има бели топчиња. Тоа значи дека во овој случај сме готови. Другиот случај е сите одговори да се да. Истото расудување како и претходно покажува дека во кутијата 2 има бело топче, со што сме готови.

Наредниот проблем е проблем од IMO кој според многу важи за еден од потешките проблеми кои биле некогаш на IMO. Сепак оставаме на вас да одлучите за тоа.

Пример 5. (IMO 2009, Problem 6) Нека n е ненегативен цел број. Скакулец скока по реланата оска. Тој почнува од точката 0 и прави $n+1$ скокови на десно со различни позитивни цели должини $a_1, a_2, ..., a_{n+1}$ во произволен редослед. Нека M е множество од n природни броеви во интервалот $(0,s)$, каде $s = a_1 + a_2 + ... + a_{n+1}$. Докажи дека скакулецот може да ги избере своите скокови така што тоа никогаш да не скокне во точка од множеството M.

Решение. Ќе конструираме алгоритам со помош на индукција и теоремата за екстреми. Случајот $n = 1$ е тривијален, па претпоставуваме дека тврдењето е точно за $1, 2, ..., n-1$. Претпоставуваме дека $a_1 < ... < a_n$. Нека m е најмалиот елемент од M. Ги разгледуваме следните случаи:

Случај 1. $m < a_{n+1}$: Ако a_{n+1} не припаѓа на M, тогаш скакулецот го прави првиот скок со големина a_{n+1}. Сега проблемот се сведува на низата $a_1, a_2, ..., a_n$ и множеството $M \setminus \{m\}$, што директно потоа следува од индукцијата. Нека сега претпоставиме дека $a_{n+1} \in M$. Ги разгледуваме следните n парови: $(a_1, a_1 + a_{n+1}), ..., (a_n, a_n + a_{n+1})$. Сите броеви од овие парови кои се во M припаѓаат на $(n-1)$-елементното множество $M \setminus \{a_n\}$, па најмалку еден од овие парови, да речеме $(a_k, a_k + a_n)$ ги има двата члена надвор од множеството M. Ако првите два скока на скакулецот се a_k и $a_k + a_{n+1}$, тој има скокнато најмалку два члена на множеството M : m и a_{n+1}. Па има најмалку $n-2$ елементи од M кои мора да ги скокне и $n-1$ скокови, па со помош на индукција сме готови.

Случај 2. $m \geq a_{n+1}$: Да забележиме дека е еквивалентно да го решаваме проблемот во обратен правец: почнувајќи од $s = a_1 + a_2 + ... + a_{n+1}$ да стигнеме до 0 без да застанеме на елемент од M. Од индуктвната претпоставка, скакулецот може да почне од s и да направи n скокови со големини $a_1, a_2, ..., a_n$ на лево и да ги избегне сите точки од $M \setminus \{m\}$. Ако го скока и m, тогаш сме готови со задачата, бидејќи сега може да се направи скок со големина a_{n+1} и да се достигне 0. Па, претпоставуваме дека после правењето на скокот со должина a_k скакулецот застанал на m. Ако го смениме скокот a_k со скокот a_n, тогаш ќе го прескокне m и сите понатамошни скокови ќе бидат надвор од M, бидејќи m е елемент од M кој е најлево позициониран од сите елементи на множеството M.

Користена литература

1. Daniel A. Greene, Donald Knuth, mathematics for the Analysis of Algorithms, Birkhauser, 1982;
2. Oded Goldreich, Computational Complexity: A Conceptual Perspective, Cambridge University Press, 2010;
3. Jason Fagone, Teen Mathletes do battle at algorithm Olympics, 2010;

4. Pranav A. Sriram, Olympiad Combinatorics, 2014;
5. Donald Knuth, Structured Programming with go-to Statements, Computing Surveys (ACM) 6 (4), 2013;
6. Donald Knuth, The art of Computer Programming, Addison-Weley, 1968.