Algoritmi şi Structuri de Date I. Seminar 1: Rezolvarea algoritmică a problemelor

- De la ce pornim? Ce se dă şi ce se cere?
- Putem abstractiza problema? Este problema similară unor probleme pe care le-am rezolvat anterior?
- Este vreo diferenţă între modul în care rezolvăm problemele în matematică şi abordarea din informatică?

Problema 1 (*Examen admitere la Informatică.*) La examenul de bacalaureat am obţinut media generală 7.50, nota la proba de Matematică fiind 7.25 iar la Informatică 8.75. Care este nota minimă pe care ar trebui să o obţin la proba scrisă a examenului de admitere pentru ca media de admitere să fie cel puţin 5?

Extras din Metodologia de admitere la specializările de Informatică:

> La domeniul Informatică media de admitere se calculează ca $N = (N3 + \max\{N1, N2\})/2$, unde $N1$ este media generală la examenul de bacalaureat, $N2$ este nota la proba de matematică sau informatică de la examenul de bacalaureat (se alege cea mai mare dintre ele, iar lipsa notei se echivalează cu nota 4) iar $N3$ este nota obţinută la proba scrisă de admitere la matematică sau informatică.

Indicaţie. Datele de intrare sunt: $N1$ (media generală la examenul de bacalaureat), $N2m$ (nota la proba de Matematică de la bacalaureat), $N2i$ (nota la proba de Informatică de la bacalaureat), $Nmin$ (nota minimă de promovare a examenului de admitere). Se cere să se determine cea mai mică valoare a lui $N3$ care asigură că N este mai mare decât $Nmin$. Din punct de vedere matematic problema este echivalentă cu rezolvarea inecuaţiei: $(N3 + \max\{N1, N2m, N2i\})/2 \geq Nmin$.

Prin rezolvarea inecuaţiei se obţine $N3 \geq 2Nmin - \max\{N1, N2m, N2i\}$. Din punct de vedere matematic, problema este rezolvată. Din punct de vedere informatic este necesară descrierea secvenţei de prelucrări care permite calculul marginii inferioare determinate matematic, ceea ce presupune următoarea secvenţă de prelucrări:

(i) determinarea maximului dintre trei valori;

(ii) calculul produsului dintre 2 şi $Nmin$;

(iii) calculul diferenţei.

Cum ar putea fi rezolvată problema fără să recurgem la rezolvarea inecuaţiei? (presupunem că nu ştim deloc matematică şi nu am auzit de inecuaţii).

Problema 2 (*Inmulţirea à la russe.*) Se consideră următoarea metodă de înmulţire (numită înmulţirea "à la russe") a două numere naturale nenule x şi y: "Se scrie x alături de y (pe aceeaşi linie). Se împarte x la 2 şi câtul impărţirii se scrie sub x (restul se ignoră deocamdată). Se înmulţeşte y cu 2 iar produsul se scrie sub y. Procedeul continuă construindu-se astfel două coloane de numere. Calculele se opresc în momentul în care pe prima coloană se obţine valoarea 1. Se adună toate valorile de pe coloana a doua care corespund unor valori impare aflate pe prima coloană."

Exemplu. Fie $x = 13$ şi $y = 25$. Succesiunea de rezultate obţinute prin aplicarea operaţiilor de mai sus este:

x	y	rest	factor multiplicare y
13	$\boxed{25}$	1	2^0
6	50	0	2^1
3	$\boxed{100}$	1	2^2
1	$\boxed{200}$	1	2^3
	325		

a) Prelucrarea descrisă prin metoda de mai sus se termină întotdeauna după un număr finit de paşi ? Ce se întâmplă în cazul în care una dintre valori este egală cu 0?

b) Metoda de mai sus conduce întotdeauna la produsul celor două numere (cu alte cuvinte este o metodă corectă de înmulţire)?

c) Câte operaţii de înmulţire cu 2 sunt necesare? Depinde acest număr de ordinea factorilor?

Indicaţie.

a) Ca urmare a împărţirilor succesive la 2 valorile de pe prima coloană descresc până se ajunge la un cât egal cu 1 (în ipoteza că x este nenul). Prin urmare prelucrarea este finită. Dacă x este egal cu 0 metoda nu poate fi aplicată ca atare, însă dacă y este 0 ea conduce la un rezultat corect.

b) Corectitudinea prelucrării derivă din faptul că metoda realizează de fapt conversia în baza 2 a primului număr (cifrele reprezentării în baza doi sunt resturile obţinute prin împărţirile succesive la 2) iar produsul se obţine prin înmulţirea succesivă a celui de al doilea număr cu puteri ale lui 2 şi prin însumarea acelor produse care corespund unor cifre nenule în reprezentarea binară a primului număr.

c) Numărul înmulţirilor cu 2 este cu 1 mai mic decât numărul de cifre ale reprezentării binare a lui x, adică $[\log_2 x]$. Evident numărul înmulţirilor depinde de ordinea factorilor fiind mai mic dacă împărţirile la 2 se efectuează asupra celui mai mic număr.

Link: http://mathforum.org/dr.math/faq/faq.peasant.html

Problema 3 (*Determinarea părţii întregi.*) Propuneţi o metodă bazată pe operaţii de adunare, scădere şi comparare pentru determinarea părţii întregi a unui număr real. Partea întreagă (numită uneori parte întreagă inferioară şi notată pentru un număr real x cu $\lfloor x \rfloor$) a unui număr real este definită ca fiind cel mai mare număr întreg mai mic decât numărul real respectiv. De exemplu, partea întreagă a lui 2.4 este 2, iar pentru -2.4 este -3.

Indicaţie. Se va ţine cont de semnul numărului. În cazul în care este pozitiv se scade succesiv valoarea 1 până când se ajunge la o valoare mai mică strict decât 1. Numărul scăderilor efectuate indică valoarea părţii întregi. Pentru numere negative se adună succesiv 1 până se obţine o valoare mai mare sau egală cu 0. Opusul numărului de adunări reprezintă valoarea părţii întregi.

Exemplu.

$x > 0$		$x < 0$	
x	contor	x	contor
3.25	0	-3.25	0
2.25	1	-2.25	-1
1.25	2	-1.25	-2
0.25	$\boxed{3}$	-0.25	-3
		0.75	$\boxed{-4}$

Problema 4 (*Problema identificării monedei mai uşoare.*) Se consideră un set de n monede identice, cu excepţia uneia care are greutatea *mai mică* decât a celorlalte. Folosind o balanţă simplă (care permite doar compararea greutăţii monedelor plasate pe cele două talere) să se identifice moneda cu greutatea mai mică folosind cât mai puţine comparaţii.

Indicaţie. O primă variantă este aceea prin care se selectează la întâmplare două monede şi se pun pe balanţă. Dacă una dintre ele are greutatea mai mică atunci este chiar moneda căutată. Daca ambele au aceeaşi greutate se păstrează una dintre ele pe un taler al balanţei iar pe celălalt taler se pun succesiv monedele rămase. Prelucrarea se opreşte în momentul în care se găseşte o monedă cu greutatea mai mică. Numărul de comparaţii este cel mult $n-1$.

O altă variantă este cea în care se împarte setul de monede în două subseturi (fiecare va avea $n/2$ elemente, dacă n e par, respectiv $(n-1)/2$ daca n este impar; în cazul în care n este impar se pune o monedă deoparte) care se pun pe cele două talere ale balanţei. Dacă subseturile au aceeaşi greutate (ceea ce se poate întâmpla doar daca n este impar) atunci moneda pusă deoparte este cea căutată. În caz contrar se aplică acelaşi procedeu subsetului de greutate mai mică şi se continuă până se ajunge la un set de două sau trei monede. În acest moment este suficient să se mai efectueze o singură cântărire. Numărul de cântăriri este cel mult $\lfloor \log_2 n \rfloor$. Este adevărat acest rezultat şi pentru cazul în care se ştie doar că una dintre monede este diferită dar nu se ştie dacă este mai uşoară sau mai grea decât celelalte?

Problema 5 (*Problema clătitelor.*) La un restaurant bucătarul a pregătit clătite (americane) pe care le-a aşezat pe un platou sub forma unei stive. Din păcate nu toate clătitele au acelaşi diametru astfel că stiva nu arată estetic. Chelnerul ia platoul şi având la dispoziţie o spatulă reuşeşte să aranjeze cu o singură mână clătitele astfel încât să fie în ordinea descrescătoare a diametrelor (cea mai mare clatită pe platou iar cea mai mică în vârful stivei). Cum a procedat? Descrieţi problema într-o manieră abstractă şi propuneţi un algoritm de rezolvare.

Indicaţie. Rearanjarea se face efectuând doar mişcări de răsturnare a unui "set" de clătite dintre cele aflate in partea de sus a stivei. Problema este identică cu cea a ordonării descrescătoare a unui şir de valori prin inversarea ordinii elementelor unor subsecvenţe de la sfârşitul şirului.

Exemplu. Pentru a ordona descrescător şirul $(5, 3, 4, 1, 6, 2)$ se aplică următoarea secvenţă de prelucrări în care subsecvenţa care se "răstoarnă" este încadrată:

$5\ 3\ 4\ 1\ \boxed{6\ 2}$
$\boxed{5\ 3\ 4\ 1\ 2\ 6}$
$6\ \boxed{2\ 1\ 4\ 3\ 5}$
$6\ 5\ 3\ \boxed{4\ 1\ 2}$
$6\ 5\ \boxed{3\ 2\ 1\ 4}$
$6\ 5\ 4\ \boxed{1\ 2\ 3}$
$6\ 5\ 4\ 3\ 2\ 1$

Metoda constă în determinarea valorii maxime din secvenţă şi inversarea ordinii subsecvenţei care începe cu ea pentru a fi adusă pe ultima poziţie în şir. După aceea se răstoarnă întregul şir, astfel că valoarea maximă ajunge pe prima poziţie. Se aplică aceeaşi metodă pentru subsecvenţa care începe cu al doilea element şi se continuă până se ajunge la o subsecvenţă constituită dintr-un singur element.

Problema 6 (*Problema partiţionării echilibrate.*) Considerăm un set de programe pentru care se cunosc estimări ale timpilor de execuţie şi care trebuie executate utilizând două calculatoare similare. Propuneţi o modalitate de distribuire a execuţiei programelor pe cele două calculatoare astfel încât încărcarea acestora să fie cât mai echilibrată.

Indicaţie. Problema poate fi reformulată (abstractizată) astfel: se consideră o mulţime de numere pozitive, $A = \{a_1, a_2, \ldots, a_n\}$, $n > 1$ şi se cere să se determine două submulţimi disjuncte B şi C astfel încât $B \cup C = A$ şi $|\sum_{a \in B} a - \sum_{a \in C} a|$ este minimă. Cea mai simplă metodă este cea a "forţei brute" prin care se generează toate perechile de submulţimi (B, C) şi se alege cea care minimizează diferenţa specificată. Numărul de partiţii distincte (fără a ţine cont de ordinea calculatoarelor) este $2^{n-1} - 1$.Pentru n mare, numărul de partiţii ce trebuie testate devine mare (pentru $n = 10$ este 511 iar pentru $n = 100$ este de ordinul 10^{29}). E evident că în astfel de situaţii metoda forţei brute nu este eficientă, astfel că trebuie căutate metode mai puţin costisitoare (astfel de metode vor fi discutate în seminariile viitoare).

Problema 7 (*Problema acoperirii.*) Se consideră o mulţime cu n elemente (de exemplu $U = \{1, 2, \ldots, n\}$) şi un set S de submulţimi ale acesteia. Se pune problema determinării unei acoperiri minimale a mulţimii U folosind submulţimi din setul S (o acoperire minimală este un subset $A \subset S$ cu proprietatea că $\cup_{s \in A} s = U$ iar numărul de elemente din A este minim). De exemplu pentru $n = 5$ se poate considera setul de submulţimi $S = \{S_1, S_2, S_3, S_4\}$ cu $S_1 = \{1, 3, 5\}$, $S_2 = \{1, 4\}$, $S_3 = \{2, 4\}$, $S_4 = \{2, 5\}$. In acest caz acoperirea minimală se obţine selectând prima dată submulţimea cu cele mai multe elemente (S_1), eliminând din U elementele submulţimii alese (U devine $\{2, 4\}$) şi continuând prin selectarea unei submulţimi care conţine elemente din U (în acest exemplu ar fi S_3). Garantează strategia de a alege la fiecare etapă cea mai numeroasă mulţime care acoperă elementele rămase obţinerea unei soluţii optime? Dacă nu, propuneţi un contraexemplu.

Indicaţie. Strategia de a alege tot timpul cea mai mare submulţime nu garantează obţinerea unei soluţii optime. Pentru $n = 6$ un contraexemplu este $S_1 = \{1, 3, 5\}$, $S_2 = \{2\}$, $S_3 = \{4\}$, $S_4 = \{6\}$, $S_5 = \{1, 2\}$, $S_5 = \{3, 4\}$, $S_5 = \{5, 6\}$.

Probleme suplimentare.

1. Se pune problema determinării valorii în baza 10 a unui număr natural pornind de la şirul cifrelor sale binare. Propuneţi o metodă bazată doar pe operaţii elementare (adunare, scădere, înmulţire - ridicarea la putere nu este considerată operaţie elementară). Se urmăreşte ca numărul operaţiilor implicate să fie cât mai mic.

2. Se consideră un număr constituit din 10 cifre distincte, diferit de 9876543210. Să se determine numărul care îl succede în şirul crescător al tuturor numerelor naturale ce conţin 10 cifre distincte.

3. Se consideră un şir de valori nenule (pozitive şi negative). Să se transforme şirul astfel încât toate valorile negative să le preceadă pe cele pozitive iar numărul operaţiilor efectuate să fie cât mai mic.

4. Se consideră un şir cu $n - 1$ valori distincte din mulţimea $\{1, 2, \ldots, n\}$. Să se determine, folosind un număr cât mai mic de comparaţii, valoarea care lipseşte din şir.

5. Se consideră următorul joc cu doi jucători bazat pe două stive de monede: una conţinând m monede şi cealaltă conţinând n monede. La fiecare etapă jucătorul care este la rând poate ridica o monedă dintr-una dintre stive sau câte o monedă din ambele stive. Câştigă jucătorul care ridică ultima/ultimele monede. Propuneţi un algoritm care pentru o pereche de valori (n, m) decide dacă pentru jucătorul care face prima mutare există o strategie câştigătoare.

6. Căutaţi pe internet (fragmente din) cartea *How to solve it?* a matematicianului George Polya şi parcurgeţi partea referitoare la etapele care e bine să fie parcurse în procesul de rezolvare a unei probleme.

7. Se consideră următorul joc, calculatorul alege în mod aleator un număr întreg x cuprins între 1 şi n pe care jucătorul doreşte să îl descopere. La fiecare rundă jucătorul va încerca să ghicească numărul. În cazul în care nu nimereste calculatorul va da un indiciu dacă numărul încercat este mai mic sau mai mare decat x. Se urmăreşte ca numărul de încercări să fie cat mai mic.