

Pregătire pentru examenul de Bacalaureat și examenul de admitere la Facultatea de Matematică și Informatică

Probleme propuse pentru sesiunea 3

27 ianuarie 2018

INFORMATICĂ

Tema 3: Algoritmi de prelucrare a tablourilor unidimensionale: construire, parcurgere, transformare, căutare.

1. *Parcurgere și construire tablouri.* Se consideră două tablouri A și B de lungimi egale, având fiecare n elemente numere întregi. Să se scrie câte o funcție/procedură care:
 - a. Calculează media elementelor din A care corespund unor valori negative în B
 - b. Construiește un tablou C care conține pe poziția i media aritmetică a elementelor corespondente din A și B .
2. *Conversii între baze de numerație (utilizând tablouri).*
 - a. Se consideră că numărul natural nenul n este reprezentat în baza 10 și se dorește construirea numărului m care reprezintă aceeași valoare însă în baza b ($2 \leq b \leq 9$). Numărul m va fi reprezentat printr-un șir de caractere.
 - b. Se consideră un tablou B cu $4k$ elemente care conține reprezentarea în baza 2 a unui număr natural și se pune problema construirii tabloului H cu k elemente care conține reprezentarea în baza 16. De exemplu pentru $B=10110011$ obține $H=B3$.
 - c. Se consideră un șir de caractere care conține reprezentarea în baza 16 a unui număr (alfabetul folosit este '0', '1', ..., 'A', ..., 'F') și se pune problema determinării valorii corespunzătoare în baza 10. De exemplu pentru $B9F$ se obține 2975.
3. *Reprezentarea în complement față de 2.* Numerele întregi negative sunt reprezentate intern prin regula complementului față de 2 care constă în aplicarea următoarelor etape: i) se determină reprezentarea în binar a numărului fără semn și se completează toți biții (un bit 0 se transformă în 1, iar un bit 1 se transformă în 0); ii) se adună 1 la numărul anterior (folosind regulile de adunare în baza 2). Ceea ce se obține este reprezentarea internă a numărului întreg negativ. Dacă se dă o reprezentare internă (0xA2B3C4D5) prin examinarea bitului de semn (cel mai din stanga) se decide dacă numărul este negativ sau nu. Dacă numărul este negativ se vor aplica aceleași reguli pentru a determina valoarea numărului. Să se scrie o funcție care să compare două numere întregi pe 32 de biți pentru care se cunoaște reprezentarea internă, în hexazecimal. Funcția va returna
 - a. -1 dacă primul număr este mai mic decât al doilea
 - b. 0 dacă sunt egale

- c. 1 dacă primul număr este mai mare decât al doilea.
4. Definiți un tablou cu numele de numere reale (tip double) de dimensiune 100. Definiți un ciclu ce va popula elementele tabloului cu următoarea secvență:
 $1/(2*3*4)$, $1/(4*5*6)$, $1/(6*7*8)$, ... până la $1/(200*201*202)$
Presupunând că numele tabloului este `data`, parcurgeți tabloul pentru a calcula următoarea expresie:
`data[0]-data[1]+data[2]-data[3]+... -data[99]`
Înmulțiți rezultatul obținut anterior cu 4.0 și adăugați 3.0 și afișați rezultatul. Recunoașteți valoarea obținută?
5. *Analiză zaruri.* Bogdan are 2 zaruri cu câte 6 fețe fiecare, unul Albastru și unul Rosu. Le aruncă pe o masă de 100 de ori și își notează valorile obținute pentru fiecare zar în parte (în câte un tablou cu 100 de elemente pentru fiecare zar). Să se decidă care dintre cele 2 zaruri este cel mai echilibrat zar? Cel albastru sau cel roșu? (Notă: Se va folosi formula entropiei lui Shannon pentru a determina cantitatea de informație existentă în fiecare zar. Zarul cu entropia mai mare este considerat mai echilibrat. Dacă (f_1, f_2, \dots, f_6) reprezintă frecvențele relative asociate valorilor obținute la aruncarea unui zar regula de calcul a entropiei este $-f_1 \cdot \log_2(f_1) - f_2 \cdot \log_2(f_2) - \dots - f_6 \cdot \log_2(f_6)$)
6. *Gigel, Ionel și Călin*, trei studenți de la Informatică au pornit un StartUp de succes împreună. Fiecare știe cateva limbaje sau tehnici de programare: *Gigel știe n limbaje*, *Ionel știe m* iar *Calin, k*. Înainte de a porni startup-ul cu propriul lor proiect pentru a face bani s-au hotărât să participe la orice proiect pe care l-ar putea face de pe freelancer.com în orice combinație posibilă. Fiecare astfel de proiect necesită o mulțime de cunoștințe descrise foarte clar în descrierea sa. Să presupunem că se cunosc cerințele pentru 30 de proiecte, să se scrie un program care să stabilească alocarea studenților la proiecte în așa fel încât să nu se suprapună competențele (pentru fiecare proiect se stabilește dacă vor intra singuri, câte doi sau toți trei împreună).
- Exemplu: Gigel știe {C, C#, OOP ,și Java}, Ionel {C, Python, C++, OOP} și Călin {C++, OOP, Ruby}. Dacă se cunosc cerințele pentru
- Proiect 1: {OOP, C, Ruby, Python} → este nevoie de toți 3 pentru acest proiect
- Proiect 2: {C#, Java} → Gigel îl poate face singur
7. *Planificare service auto.* Într-un service auto există 5 elevatoare și 5 mecanici. Într-o lună se repară în acel service, în medie, 200 de mașini. Seful de service, Gigel, este cel care planifică fiecare mașină știind pentru fiecare intervenție când intră în service și cât durează reparația. Se cunosc datele din planificare cand intră fiecare mașină și cât durează intervenția (vor fi generate random), și mai știm că Gigel atunci când planifică se uită pe rând care elevator este liber și îl marchează în planificare. Dacă nu are niciunul disponibil atunci împreună cu clientul alege o altă dată și oră disponibilă. Întrebarea la care vrem să răspunem este care din cele 5 elevatoare va fi folosit cel mai mult în luna ce urmează și care vor fi mașinile reparate în ordinea cronologică a reparațiilor.

Exemplu: presupunem că în loc de 20 zile a câte 8 ore pe zi ne gândim la 160 de ore pe lună. Ca intrare vom primi 2 seturi de numere care specifică ora la care începe și cât durează intervenția pentru o mașină (1,1,5,10,7) și (4,6,4,3,4):

Mașina Ora start Durata

1	1	4
2	1	6
3	5	4
4	10	3
5	7	4

vom planifica astfel

1,1+4 → 5, 4+5 → 10, 10+3

1,1+6 → 7, 7+4

8. *Clasament WTA*. În acest moment (Ianuarie 2018) în clasamentul WTA la tenis feminin primele 10 jucătoare au următoarele puncte:

1	[ROU] Simona Halep	6425
2	[DEN] Caroline Wozniacki	6095
3	[ESP] Garbiñe Muguruza	6050
4	[UKR] Elina Svitolina	5785
5	[USA] Venus Williams	5568
6	[CZE] Karolina Pliskova	5445
7	[LAT] Jelena Ostapenko	4901
8	[FRA] Caroline Garcia	4385
9	[USA] Coco Vandeweghe	3204
10	[GBR] Johanna Konta	3185

Presupunem că în fiecare lună va fi câte un concurs de GrandSlam și știm că pentru GrandSlam se pot câștiga următoarele puncte

	W	F	SF	QF	R16	R32	R64	R128	Q	Q3	Q2	Q1
Points	2000	1300	780	430	240	130	70	10	40	30	20	2

(W=winner, F=Finalist, ...)

Fiecare dintre jucătoarele din top 10 ar putea să câștige oricare meci și oricare turneu. Să presupunem că se cunoaște pentru următoarele 10 concursuri în ce fază ajunge fiecare jucătoare (în implementare valorile pot fi alocate aleator) să se determine dacă peste 10 luni Simona Halep va mai fi numărul 1 sau nu.

9. *Suffix și prefix comun*. Fiind date două tablouri (a_1, a_2, \dots, a_m) și (b_1, b_2, \dots, b_n) , cu numere întregi, să se stabilească numărul maxim de elemente comune aflate la sfârșitul lui a și începutul lui b (de exemplu $a=(1,4,3,5,6,7,2)$ și $(6,7,2,5,1,8)$ au 3 elemente comune, pe 6, 7 și 2).

10. *Date de naștere.* Se consideră o listă care conține datele de naștere ale locuitorilor unui oraș (lista poate fi reprezentată printr-un tablou conținând structuri sau prin 3 tablouri corespunzătoare celor 3 componente ale date: (zi, luna, an)). Să se determine:
- cea mai „mică” dată de naștere (corespunzătoare celui mai vârstnic locuitor)
 - cea mai „mare” dată de naștere (corespunzătoare celui mai tânăr locuitor)
 - cea/cele mai frecventă dată de naștere (cea/cele care apare/apar de cele mai multe ori în listă)
 - dacă există vreo zi în perioada (1 ianuarie 1918 – 31 decembrie 2017) în care nu e născut nici un locuitor.