Tema nr. 1

1. Să se găsească cel mai mic număr pozitiv $u > 0$, de forma $u = 10^{-m}$ care satisface proprietatea:

$$1 +_c u \neq 1$$

unde prin $+_c$ am notat operația de adunare efectuată de calculator. Numărul u se numește ***precizia mașină***.

2. Operația $+_c$ este ***neasociativă***: fie numerele $x=1.0$, $y = u$, $z = u$, unde u este precizia mașină calculată anterior. Să se verifice că operația de adunare efectuată de calculator este neasociativă:

$$(x +_c y) +_c z \neq x +_c (y +_c z).$$

Să se găsească un exemplu pentru care operația de înmulțire $\times_c$ este neasociativă.

3. **Înmulțirea matricelor booleene** (***Algoritmul celor patru ruşi***) **-** Aho, A. V., Hopcroft, J. E., & Ullman, J. D. (1976), The Design and Analysis of Computer Algorithms, Addison-Wesley.

Fie $A, B \in \{0,1\}^{n \times n}$ două matrice pătratice de dimensiune n cu elemente în $\{0,1\}$. Se cere să se calculeze matricea produs $C = A \cdot B \in \{0,1\}^{n \times n}$. Operațiile de adunare şi înmulțire pe $\{0,1\}$ sunt următoarele:

+	0	1
0	0	1
1	1	1

×	0	1
0	0	0
1	0	1

Presupunând că n se divide la $\log n$, se împart matricele A şi B în submatrice de dimensiune $n \times \log n$ respectiv $\log n \times n$ astfel:

A =

A_1	A_2	...	$A_{n/\log n}$

B =

B_1
B_2
⋮
$B_{n/\log n}$

Matricea produs ***C*=*A*•*B*** se poate calcula, folosind partiționarea de mai sus, astfel:

$$C = \sum_{i=1}^{\frac{n}{\log n}} A_i B_i \in \{0,1\}^{n\times n}.$$

Fiecare linie din matricea produs $C_i = A_i B_i$ este obținută sumând anumite linii ale matricei ***B***, (alegerea liniilor din ***B*** care se sumează este în funcție de elementele nenule ale liniilor din matricea ***A***).

$$A_i = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad B_i = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

$$C_i = A_i B_i = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} \textbf{linia 1 din } B + \textbf{linia 3 din } B \\ \textbf{linia 2 din } B + \textbf{linia 3 din } B \\ \textbf{linia 1 din } B + \textbf{linia 2 din } B \\ 0 \\ \textbf{linia 1 din } B \\ \textbf{linia 1 din } B + \textbf{linia 2 din } B + \textbf{linia 3 din } B \\ \textbf{linia 2 din } B \\ \textbf{linia 3 din } B \end{bmatrix}$$

Ideea algoritmului celor 4 ruși pentru a obține matricea booleană $C_i = A_i B_i$ este de a calcula toate cele ***n*** variante posibile de sume ale liniilor matricei ***B*,** liniile matricei C_i se găsesc alegând varianta corespunzătoare liniei din A_i care se folosește pentru calculul liniei respective din C_i.

Se notează cu $m = \lfloor \log n \rfloor$ și $p = \left\lceil \frac{n}{m} \right\rceil$ unde cu $\lfloor x \rfloor$ am notat partea întreagă a numărului x, i.e. cel mai mare număr întreg $\le x$ iar $\lceil x \rceil$ este cel mai mic număr întreg $\ge x$.

Pentru un vector boolean $v \in \{0,1\}^n$ $v = (v_1, v_2, ..., v_n)$, $v_k \in \{0,1\}$ se notează cu ***NUM(v)*** numărul întreg care are reprezentarea în baza 2 $\overline{v_n v_{n-1} \cdots v_1}$.

Algoritmul celor 4 ruși de calcul al matricei produs $C=AB$

1. Se împarte matricea A în submatricele $A_1, A_2, \ldots, A_p$ unde coloanele matricei A_i sunt coloanele matricei A de la $m(i-1)+1$ la mi pentru $1 \leq i < p$ iar A_p conține coloanele rămase din A la care se adaugă suplimentar, dacă este necesar, coloane de 0-uri pentru ca și submatricea A_p să aibă m coloane.

2. Se împarte matricea B în submatricele $B_1, B_2, \ldots, B_p$ unde liniile matricei B_i sunt liniile matricei B de la $m(i-1)+1$ la mi pentru $1 \leq i < p$ iar B_p conține liniile rămase din B la care se adaugă suplimentar, dacă este necesar, linii de 0-uri pentru ca și submatricea B_p să aibă m linii.

3. Se calculează matricele produs $C_i = A_i B_i$ astfel:

for $i=1,p$

// se calculează toate sumele posibile ale liniilor matricei B_i: $b_1^{(i)}, b_2^{(i)}, \ldots, b_m^{(i)}$

$sum\_linii\_B(0) = \underbrace{[0\ 0 \cdots 0]}_{n};$

for $j=1,2^m-1$

fie k astfel încât $2^k \leq j < 2^{k+1}$;

$sum\_linii\_B(j) = sum\_linii\_B(j-2^k) + b_{k+1}^{(i)}$;

linia r din matricea produs C_i este $sum\_linii\_B(NUM(a_r^{(i)}))$ unde $a_r^{(i)}$ este linia r a matricei A_i, $1 \leq r \leq n$.

4. Matricea produs C se obține sumând toate matricele C_i obținute la pasul 3.

Fie $A, B \in \mathbb{R}^{n \times n}$ două matrice reale pătratice de dimensiune n. Elementele matricei produs $C = A * B \in \mathbb{R}^{n \times n}$ se calculează folosind formula clasică:

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj} \ , \quad i, j = 1, \ldots, n.$$

Folosind modul clasic de calcul al produsului a două matrice, complexitatea algoritmului este de ordinul $\mathcal{O}(n^3)$ iar metoda celor 4 ruși are o complexitate de calcul de ordinul $\mathcal{O}(n^3 / \log n)$.