# Tema AA

22.10.2012

Rezolvarea unei teme va conţine, în partea superioară a primei pagini, următorul tabel, pentru trecerea punctajului. Capul de coloana are semnificatia $\langle$*numar sectiune*$\rangle$.$\langle$*numar exercitiu*$\rangle$.

| 1.1 | 1.2 | ... | 2.1 | ... | $m.n$ |
|---|---|---|---|---|---|
| | | | | | |

## Cuprins



*Nota*: Punctajul total este 110.

# 1 Notatii de complexitate

1. (4p) Consideram ca $A$ este un algoritm care ruleaza in timp $T_A(n) = 1000 \cdot n^2$ pe masina $M$, si $B$ este un algoritm care ruleaza in timp $T_B(n) = 2^n$ pe aceeasi masina. Care este cea mai mare valoare pentru $n$, pentru care algoritmul $B$ este mai rapid decat $A$ ?

2. (6p) Treceti in dreptul fiecarei intrari din tabelul de mai jos cea mai precisa relatie dintre $f$ si $g$, alegand dintre $O, \Theta, \Omega, o, \omega$:

| $f(n)$ | $g(n)$ | $f(n) = \dots(g(n))$ |
|---|---|---|
| $n^3 + 3n + 1$ | $n^4$ | |
| $\log n$ | $n$ | |
| $n \cdot 2^n$ | $3^n$ | |
| $n$ | $\log^5 n$ | |
| $\log n$ | $\log n^2$ | |
| $100n + \log n$ | $n + \log^2 n$ | |

3. (5p) Scrieti un program eficient care verifica daca o lista simplu inlantuita contine cicluri. Folositi notatii asimptotice pentru a descrie cat mai precis timpul de executie al programului.

4. (5p) Calculaţi rezultatele următoarelor operaţii, utilizând notaţii asimptotice cât mai precise:

   (a) $\omega(f(n)) + \Theta(f(n)) = \dots$

   (b) $\omega(f(n)) \cup o(f(n)) = \dots$

   (c) Dacă $f(n) = \Omega(n^2)$ şi $g(n) = O(n^3)$, atunci $f(n)/g(n) = \dots$

   (d) Dacă $f(n) = o(n^2)$ şi $g(n) = \Theta(n^3)$, atunci $f(n) \cdot g(n) = \dots$

   (e) Dacă $f(n) = \Theta(n^3)$ şi $g(n) = o(n^2)$, atunci $f(n)/g(n) = \dots$

# 2 Recurente

1. (6p) Identificati solutiile recurentelor de mai jos, folosind metoda arborilor iterativi. Demonstrati corectitudinea solutiei folosind metoda substitutiei.

   - $A(n) = 2A(n/4) + \sqrt{n}$
   - $B(n) = 3B(n/3) + n^2$
   - $C(n) = C(n/2) + C(n/3) + C(n/4) + n$

2. (8p) Rezolvati urmatoarea recurenta $T(n) = \sqrt{n} \cdot T(\sqrt{n}) + n$ folosind metodele Master si substitutiei.

# 3 Analiza amortizata

1. (6p) Consideram un vector $V$ in care operatia de inserare conduce la marirea de patru ori a capacitatii vectorului, daca acesta este plin.

   (a) Determinati costul mediu pentru o secventa $n$ operatii de inserare si stergere de elemente din vector

   (b) Folositi metoda creditelor pentru a identifica un cost amortizat per operatie

   (c) Identificati o functie de potential corespunzatoare pentru $V$, si extrageti costul amortizat. Este acelasi cu cel gasit anterior ?

2. (6p) Se considera o implementare a unei cozi FIFO (First In First Out), folosind doua stive $S_1$ si $S_2$. Stiva are operatiile PUSH si POP, iar costul fiecarei operatii este 1. Coada are operatiile ENQUEUE si DEQUEUE, implementate astfel:

```
ENQUEUE(x) {
        do PUSH(x) on S1
}
DEQUEUE() {
        if (S2 is empty) {
                POP() all elements from S1 and PUSH them in S2
        }
        do POP() on S2 and return the result
}
```

(a) Exemplificati continutul cozii FIFO (si implicit al stivelor) pentru o secventa oarecare (aleasa de voi) de operatii ENQUEUE si DEQUEUE;

(b) Identificati costul mediu pentru o secventa de operatii ENQUEUE si DEQUEUE, folosind metoda agregarii;

(c) Identificati costul amortizat pentru operatiile ENQUEUE si DEQUEUE folosind metoda creditelor;

(d) Identificati o functie de potential pentru coada FIFO astfel incat costurile amortizate care rezulta pe baza acesteia sa fie cele identificate la punctul anterior.

## 4 Invarianti la ciclare

1. (6p) Algoritmul lui Euclid primeste doua valori intregi $A$, $B$ si calculeaza cel mai mare divizor comun:

```
b = B, a = A, r = B
while (b != 0) {
        r = a mod b
        a = b
        b = r
}
```

   Demonstrati corectitudinea Algoritmului lui Euclid folosind invarianti la ciclare.

2. (6p) Fie algoritmul HighSchool (v. cursul), de inmultire a doua numere, reprezentate sub forma de vector. Pozitia de indice $i$ din vector stocheaza cifra de rang $i$ a numarului, indicii incepand de la 0. Pentru simplitate, vom presupune ca lungimile celor doua numere sunt *egale*.

**Algoritmul 1** HighSchool($x, y, n$)

**Intrare:** $x$, $y$ — vectorii reprezentand cele doua numere
**Intrare:** $n$ — lungimea lor
**Iesire:** vectorul reprezentand produsul acestora

```
for i ← 0, 2n − 1 do
    p[i] ← 0
end for
for i ← 0, n − 1 do
    remainder ← 0
    for j ← 0, n − 1 do
        value ← x[j] · y[i] + remainder + p[j + i]
        p[j + i] ← value mod 10
        remainder ← value div 10
    end for
    p[n + i] ← remainder
end for
return p
```

Demonstrati corectitudinea algoritmului HighSchool, utilizand invarianti la ciclare.

# 5 Inductie structurala

1. (12p) **Variabile libere si legate in Quantified Boolean Formulae**

   **Definitia 1** (Quantified Boolean Formulae (QBF))**.** *Fie $V$ o multime de variabile.*

   - *daca $v$ este o variabila din $V$, atunci $v$ este o formula QBF peste $V$.*
   - *daca $\phi$ este o formula QBF, atunci $\neg\phi$ este o formula QBF.*
   - *daca $\phi$ si $\psi$ sunt formule QBF atunci $\phi \wedge \psi$ si $\phi \vee \psi$ sunt formule QBF.*
   - *daca $v$ este o variabila si $\phi$ este o formula $QBF$, atunci $\exists v\phi$ si $\forall v\phi$ sunt formule QBF peste $V$.*

   **Exemplul 1** (QBF)**.** *Fie $V = \{a, b, c\}$. Urmatoarele sunt formule QBF:*

   $$\begin{array}{ll} (a) & \exists a\ b \wedge c \vee \neg a \\ (b) & \exists a\ b \wedge \forall c\ c \vee a \\ (c) & \exists a\ \forall c\ b \wedge c \vee a \end{array}$$

   **Observatia 1** (Cuantificatori)**.** *Intuitiv, formula $\forall v\phi$ este adevarata daca $\phi$ este adevarata pentru orice valoare asociata lui $v$ iar formula $\exists v\phi$ este adevarata daca $\phi$ este adevarata pentru o valoare oarecare a lui $v$.*

   **Definitia 2** (Aparitii libere si legate ale unei variabile)**.** *Fie $v \in V$ o variabila si $\phi$ o formula QBF peste $V$. Daca variabila $v$ apare de mai multe ori in formula $\phi$, atunci notam cu $v_1, v_2, \ldots v_n$ toate aparitiile lui $v$ in $\phi$. Exemplu: In formula $a \wedge \exists a\ c \wedge a$, variabila $a$ apare de trei ori, iar* *aparitiile sunt* $a_1, a_2$ *si* $a_3$*:* $a_1 \wedge \exists a_2\ c \wedge a_3$*. O aparitie* $v_i$ *a unei variabile* $v$ *este* ***legata*** *daca: (a) aparitia lui* $v_i$ *este de forma* $\forall v_i$ *sau* $\exists v_i$*: (b) aparitia lui* $v_i$ *este precedata de o aparitie ce corespunde punctului (a). O aparitie este* ***libera*** *daca nu este legata.*

*Exemplu: aparitiile* $a_2$ *si* $a_3$ *din formula de mai sus sunt legate. Aparitia* $a_1$ *este libera.*

**Definitia 3** (Variabile libere si legate)**.** *: Spunem ca o variabila* $v \in V$ *este* ***legata*** *intr-o formula QBF peste* $V$*,* $\phi$*, daca* ***toate aparitiile*** *acesteia in* $\phi$ *sunt legate. Daca o variabila nu este legata, atunci este libera.*

*In exemplul de mai sus, variabila a nu este legata (i.e. este libera), din cauza ca* ***nu toate*** *aparitiile acesteia sunt legate.*

**Cerinta.** *Rezolvati urmatoarele:*

(a) *Definiti un TDA asociat unei formule QBF*

(b) *Definiti operatorii* $FV(f)$ *si* $BV(f)$ *care determina variabilele libere, respectiv legate dintr-o formula QBF* $f$*.*

(c) *Demonstrati prin inductie structurala faptul ca* $FV(f) \cap BV(f) = \emptyset$*, pentru orice formula QBF* $f$*.*

2. (10p) **Interpretari pentru Quantified Boolean Formulae**

**Definitia 4** (Interpretare)**.** *Fie* $V$ *o multime de variabile. O functie* $I : V \to \{0, 1\}$ *se numeste o interpretare pentru multimea* $V$*.* $I$ *asociaza fiecarei variabile din* $V$ *o valoare de adevar (true/false, 0/1).*

*Notam cu* $I[u \leftarrow b]$ *(unde* $b$ *este o valoare de adevar: 0 sau 1) o interpretare care satisface urmatoarele conditii:*

- $I[u \leftarrow b](v) = I(v)$ *daca* $u \neq v$
- $I[u \leftarrow b](v) = b$ *daca* $u = v$

*Asadar, interpretarea* $I[u \leftarrow b]$ *este interpretarea* $I$*, in care valoarea variabilei* $u$ *este* $b$*.*

**Exemplul 2** (Interpretare)**.** *Fie* $V = \{a, b, c\}$*. Functia* $I$ *astfel incat* $I(a) = 0, I(b) = 0, I(a) = 1$*, este o interpretare pentru* $V$*. In continuare, vom folosi urmatoarea notatie mai compacta:* $I = \{a \leftarrow 0, b \leftarrow 0, c \leftarrow 1\}$*. Interpretarea* $I[a \leftarrow 1]$ *este* $\{a \leftarrow 1, b \leftarrow 0, c \leftarrow 1\}$ *si* $I[d \leftarrow 0]$ *este* $\{a \leftarrow 0, b \leftarrow 0, c \leftarrow 1, d \leftarrow 0\}$*.*

**Definitia 5** (Satisfacere)**.** *Fie* $V$ *o multime de variabile,* $I$ *o interpretare pentru* $V$ *si* $\varphi$ *o formula QBF peste* $V$*. Definim notiunea de satisfacere a unei formule* $\varphi$ *de catre o interpretare* $I$ *(notata* $I \models \varphi$*) intr-o maniera recursiva, astfel:*

- $I \models v$ *(cu* $v \in V$*) daca* $I(v) = 1$
- $I \models \phi \wedge \psi$ *daca* $I \models \phi$ *si* $I \models \psi$
- $I \models \phi \vee \psi$ *daca* $I \models \phi$ *sau* $I \models \psi$
- $I \models \neg\phi$ *daca* $I \models \phi$ *nu este adevarata.*

- $I \models \forall v \phi$ *(cu* $v \in V$*) daca* $I[v \leftarrow 1] \models \phi$ *si* $I[v \leftarrow 0] \models \phi$.
- $I \models \exists v \phi$ *(cu* $v \in V$*) daca* $I[v \leftarrow 1] \models \phi$ *sau* $I[v \leftarrow 0] \models \phi$.

**Exemplul 3** (Satisfacere)**.** *Interpretarea* $\{b \leftarrow 1, c \leftarrow 1\}$ *satisface formula din Exemplul 1 (a). Interpretarea* $\{b \leftarrow 1\}$ *satisface formula din Exemplele 1 (b) si (c).*

**Cerinta.** *Demonstrati ca, pentru doua interpretari* $I_1$ *si* $I_2$ *care convin asupra tuturor variabilelor libere ale unei formule* $f$ *(i.e.* $\forall v \in FV(v)$ *avem* $I_1(v) = I_2(v)$*) ,* $I_1 \models f \iff I_2 \models f$.

3. (9p) Fie tipul de date *List*, cu elemente de tipul *T*, avand constructorii de baza:

$$\begin{array}{rll} [\,] :\rightarrow & List & \text{(Lista vida)} \\ (:) : & T \times List \rightarrow List & \text{(Adaugare)} \end{array}$$

Constructorul intern este reprezentat prin caracterul "doua puncte", ca in expresia (3:[]). Fie operatorii: *insert(x,L)* insereaza o valoare intr-o lista, inaintea primului element mai mare sau egal cu ea. *isort(L)* utilizeaza operatorul *insert* pentru a sorta prin insertie o lista.

   (a) Definiti **tipurile** si **axiomele** corespunzatoare operatorilor *insert* si *isort*.

   (b) Demonstrati, prin **inductie structurala**, ca, daca lista *L* este sortata, atunci *insert(x,L)* este o *permutare sortata* a listei *x:L*

   (c) Demonstrati, prin **inductie structurala**, ca, pentru orice lista *L*, *isort(L)* este o *permutare sortata* a acesteia.

# 6 Clase de probleme

1. (9p) Fie urmatoarea problema ***Acoperire-Para:*** *Se da un graf G=(V,E) in care gradul*[1] *fiecarui nod este par, si o valoare k. Are G o acoperire de dimensiune k ?*

   (a) Demonstrati ca ***Acoperire-Para*** este in NP;

   (b) Demonstrati ca pentru orice graf *G*, numarul nodurilor avand un grad impar, este par;

   (c) Folosind proprietatea de mai sus, demonstrati ca ***Acoperire-Para*** este NP-completa.

2. (8p) SelfishWorld este o lume populata de agenti egoisti, care interactioneaza. Notam cu $N$ multimea acestor agenti.

   Exemplu: $N = \{John, Mary\}$.

   **Agenti si actiuni** Fiecare agent poate executa una sau mai multe actiuni in SelfishWorld. Reprezentam o actiune ca o variabila booleana $v$. Daca $v = 1$, spunem ca actiunea a fost executata. Daca $v = 0$ spunem ca

[1] Gradul unui nod reprezinta numarul de muchii incidente acestuia.

actiunea nu a fost executata. Notam multimea actiunilor cu $V$. Notam cu $V_i$ multimea actiunilor pe care agentul $i \in N$ le poate efectua. Nu exista doi agenti care sa poata executa aceeasi actiune, prin urmare toate multimile $V_i$ cu $i \in N$ sunt disjuncte.

Exemplu: Fie $V = \{jr, jf, mr, mf\}$, unde $jr$ reprezinta actiunea *John merge la restaurant*, $jf$ reprezinta actiunea *John merge la fast-food*, $mr$ reprezinta actiunea *Mary merge la restaurant* si $mf$ reprezinta actiunea *Mary merge la fast-food.* Asadar $V_{John} = \{jr, jf\}$ si $V_{Mary} = \{mr, mf\}$.

**Scop** Consideram ca fiecare agent $i$ are un scop, $\phi_i$, exprimat ca o formula booleana in forma CNF peste multimea $V$.

Exemplu: $\phi_{John} = jr \wedge mr \vee jf \wedge mf$. Scopul lui John este sa mearga la acelasi local (restaurant sau fast-food) la care merge si Mary. $\phi_{Mary} = \neg jr \wedge mr \vee \neg jf \wedge mf$. Scopul lui Mary este sa mearga la un local diferit de cel la care merge John.

**Rezultat** Numim rezultat al unei interactiuni intre agenti, o interpretare $I$, prin urmare o asignare de valori de adevar, fiecarei variabile din $V$.

Exemplu: $I = \{mr \leftarrow 1, mf \leftarrow 0, jf \leftarrow 1, jr \leftarrow 0\}$ reprezinta faptul ca Mary merge la restaurant si John merge la fast-food.

**Preferinte** Avand doua interpretari $I_1$ si $I_2$, spunem ca un agent $i$ **prefera** strict un rezultat $I_1$ lui $I_2$ (notat $I_1 \succ_i I_2$) daca $I_1 \models \phi_i$ si $I_2 \not\models \phi_i$. Un agent $i$ este indiferent intre rezultatele $I_1$ si $I_2$ daca $I_1 \models \phi_i \wedge I_2 \models \phi_i$ sau $I_1 \not\models \phi_i \wedge I_2 \not\models \phi_i$.

Exemplu: Fie $J = \{mr \leftarrow 1, mf \leftarrow 0, jf \leftarrow 0, jr \leftarrow 1\}$. $I \succ_{Mary} J$ si $J \succ_{John} I$.

**Deviere** Spunem ca un agent $i$ poate **devia** de la un rezultat $I$, daca exista un alt rezultat $I*$, astfel incat: (i) $I$ si $I*$ convin asupra variabilelor din $V \setminus V_i$ (i.e. $\forall v \in V \setminus V_i \; I(v) = I*(v)$) si (ii) $I* \succ_i I$.

**Cerinta.** *Fie* $\mathfrak{I} = (N, V, (V_i)_{i \in N}, (\phi_i)_{i \in N})$ *o descriere pentru SelfishWorld, si $I$ un rezultat al unei interactiuni din SelfishWorld. Fie urmatoarea problema:* Exista un agent care poate devia in $\mathfrak{I}$ de la rezultatul $I$? *Demonstrati ca aceasta problema este NP-completa*[2].

3. (8p) Autentificarea folosind amprente se bazeaza pe modele de abstractizare a amprentelor. Un astfel de model (simplificat) este un graf neorientat, in care:

   - nodurile reprezinta puncte distinctive ale amprentei (*minutii*)
   - muchiile reprezinta legaturi intre aceste puncte

   Consideram ca doua amprente sunt compatibile, daca eliminand cel mult $n$ noduri (impreuna cu muchiile adiacente acestora) din cele doua grafuri asociate (nu neaparat acelasi), acestea devin identice. Demonstrati ca

[2] In Economie, conceptul de *scenariu stabil* este strans legat de *devieri posibile.* Un scenariu poate fi considerat stabil daca nici un agent nu poate devia, cunoscandu-se actiunile celorlalti agenti (care sunt fixate). In acest context, multimea actiunilor care produc un scenariu stabil este un *Echilibru Nash.* Pentru modelarea aleasa in acest exercitiu, problema stabilirii daca o multime de actiuni (i.e. interpretare sau rezultat) este un *Echilibru Nash* este coNP-completa.

problema compatibilitatii a doua amprente modelate folosind grafuri, in raport cu o valoare $n$, este NP-completa.