Logică și structuri discrete

Mulțimi

Marius Minea
marius@cs.upt.ro

http://cs.upt.ro/~marius/curs/lsd/

17 octombrie 2016

Ce sunt mulțimile?

Mulțimea e un concept matematic *fundamental*. Am putea spune:
O mulțime e o *colecție* de obiecte numite *elementele* mulțimii.

... dar definiția dată e *informală*: ca să fie riguroasă, ar trebui să definim precis ce e o *colecție*.

Două noțiuni distincte: *element* și *mulțime*

$x \in S$: elementul x *aparține* mulțimii S
$x \notin S$: elementul x *nu aparține* mulțimii S

Spre deosebire de liste:
Elementele unei mulțimi *nu* sunt ordonate
Un element *nu* apare de mai multe ori
(Dar: există noțiunea de *multiset*: fiecare element e caracterizat prin numărul de apariții)

Noțiunea formală de mulțime e datorată lui Georg Cantor (1879).

Vom discuta pe scurt și despre formalizarea ei.

Moduri de definire

1. Prin *enumerarea* elementelor:
   $A = \{a, b, c\}$, $\quad D = \{1, 2, 3, 6\} =$ mulțimea divizorilor lui 6

Elementele mulțimii se scriu între acolade, separate prin virgulă.

2. Printr-o *proprietate* caracteristică
   $S = \{x \mid x \text{ are proprietatea } P(x)\}$

$D(n) = \{d \in \mathbb{N} \mid n \mod d = 0\}$ (mulțimea divizorilor lui n)

Știm: mulțimea numerelor naturale $\mathbb{N}$, întregi $\mathbb{Z}$, raționale $\mathbb{Q}$, reale $\mathbb{R}$, ...

Submulțimi

A e o *submulțime* a lui B:
$A \subseteq B$ dacă fiecare element al lui A e și un element al lui B.

A e o *submulțime proprie* a lui B: $A \subset B$
dacă $A \subseteq B$ și există cel puțin un element $x \in B$ așa încât $x \notin A$.

Obs. $\in$ e o relație între un *element* și o mulțime.
$\subseteq$ (și $\subset$) sunt relații între *două mulțimi*.

Obs. Ca să demonstrăm $A \not\subseteq B$ e suficient să găsim un element $x \in A$ pentru care $x \notin B$.

Dacă $A \subseteq B$ și $B \subseteq A$, atunci $A = B$ (mulțimile sunt egale)
demonstrăm egalitatea unor mulțimi definite prin proprietățile lor

Operații de bază

Reuniunea a două mulțimi:
$A \cup B = \{x \mid x \in A \text{ sau } x \in B\}$

Intersecția a două mulțimi:
$A \cap B = \{x \mid x \in A \text{ și } x \in B\}$

Diferența a două mulțimi:
$A \setminus B = \{x \mid x \in A \text{ și } x \notin B\}$

Uzual, discutăm într-un *context*: avem un *univers* (de discurs) U al tuturor elementelor la care ne-am putea referi.

Complementul unei mulțimi (în raport cu universul U):
$A^c = \{x \in U \mid x \notin A\} = U \setminus A$ (uneori notat și $\bar{A}$)

Pot fi reprezentate prin *diagrame Venn*

Funcția caracteristică a unei mulțimi

Dacă fixăm un univers U de elemente posibile, putem reprezenta orice mulțime $S \subseteq U$ prin *funcția caracteristică* $f_S : U \to \mathbb{B}$, $f(x) = true$ dacă $x \in S$, și *false* altfel (dacă $x \notin S$)

Un limbaj funcțional poate reprezenta *date* (mulțimi) prin *funcții*

Pornim de la mulțimea cu un singur element, *a*

```
let singleton a = fun x -> x = a (*adev. doar pt. a *)
```

`singleton a` are tipul `'a -> bool`: mulțimea e o funcție

testul de element e *aplicarea funcției* la element: `m x`

```
let empty = fun _ -> false (*functia constanta *)
```

Operațiile pe mulțimi se exprimă cu operatori booleni

```
let add a m = fun x -> x = a || m x (*adauga elem *)
let union m1 m2 = fun x -> m1 x || m2 x
let inter m1 m2 = fun x -> m1 x && m2 x
let diff m1 m2 = fun x -> m1 x && not m2 x
```

Mulțimile, fundament al matematicii

Practic toată matematica poate fi formalizată în teoria mulțimilor (sau în logică, de care e strâns legată, după cum vom vedea).

Exemplu: o *pereche* (deși e ordonată!) poate fi definită ca:
$(a, b) = \{\{a\}, \{a, b\}\}$ (definiția lui Kuratowski)
cum putem extrage pe a și b fiind dată perechea (a, b) ?

Numerele naturale au fost formalizate de Peano:
0 e un număr natural
dacă n e un număr natural, $S(n)$ e un număr natural
(funcția succesor S e injectivă, și $S(n) \neq 0$ pentru orice n)

Putem să definim numerele naturale folosind mulțimi:
$0 \stackrel{def}{=} \emptyset$
$S(n) \stackrel{def}{=} n \cup \{n\}$

Paradoxul lui Russell

O formulare intuitivă (paradoxul bărbierului):

Bărbierul bărbierește exact oamenii care nu se bărbieresc singuri.
Bărbierul se bărbierește pe el însuși sau nu ?

E cauzat de presupunerea (în teoria naivă a mulțimilor) că orice predicat $P(x)$ (proprietate a unor valori) poate defini o mulțime

$$\exists y \forall x (x \in y \Leftrightarrow P(x)) \qquad (y \text{ e mulțimea definită})$$

Căutăm să obținem o echivalență între o propoziție și negația ei:
alegem $P(x) : x \notin x$ și luăm $x = y$ (în $\forall x$... putem alege orice x).
Obținem $y \in y \Leftrightarrow y \notin y$, paradox.

Sau: dacă $R = \{X \mid X \notin X\}$, mulțimea R se conține pe ea însăși?
dacă $R \in R$, pentru a satisface condiția de definiție, avem $R \notin R$.
dacă $R \notin R$, atunci R satisface condiția, și $R \in R$: paradox!

Paradoxul a pus probleme serioase formalizării logicii matematice

Paradoxul lui Russell (cont.)

Poate fi *evitat* în mai multe feluri, impunând *restricții* asupra modului în care se poate defini o mulțime.

de ex.: Nu putem defini o mulțime doar printr-o proprietate $P(x)$, trebuie să *specificăm universul* din care își poate lua elementele:

$$R = \{X \mid X \subseteq U \text{ și } X \notin X\}$$

Dacă presupunem $R \in R$, din proprietatea care definește mulțimea, rezultă $R \notin R$
(nu e un paradox, înseamnă doar că presupunerea a fost falsă).

Dacă $R \notin R$, rezultă doar că nu putem avea $R \subseteq U$ *și* $R \notin R$.

Rezultă că $\neg(R \subseteq U)$, deci R nu e o mulțime (valid definită) în universul considerat.

Teoria axiomatică a mulțimilor

O *axiomă* e o propoziție presupusă adevărată.
E un punct de plecare pentru un raționament.

Sistemele axiomatice au fost dezvoltate pentru a evita paradoxurile din *teoria naivă* a mulțimilor (cu noțiuni definite în limbaj natural)

Cel mai răspândit: sistemul *Zermelo-Fraenkel* (1907..1930).
Câteva axiome:

Axioma extensionalității:
Două mulțimi sunt egale dacă și numai dacă au aceleași elemente (dacă fiecare element al lui A e și un element al lui B, și reciproc)

$$\forall A \forall B (A = B \Leftrightarrow \forall C (C \in A \Leftrightarrow C \in B))$$

Axioma mulțimii vide (existență):
Există o mulțime care nu are niciun element

$$\exists E \forall X \neg (X \in E)$$

...

Axiome ale teoriei mulțimilor (cont.)

Axioma regularității (a fundației)
Orice mulțime nevidă are un element $x \in A$ disjunct de ea: $x \cap A = \emptyset$

$$\forall X(X \neq \emptyset) \Rightarrow \exists Y(Y \in X \wedge \neg\exists Z(Z \in X \wedge Z \in Y))$$

Rezultă că nu există un șir infinit $A_0, A_1, \ldots A_n \ldots$ astfel încăt

$$A_0 \ni A_1 \ni \ldots \ni A_n \ni \ldots$$

(altfel $\{A_0, A_1, \ldots\}$ ar fi o astfel de mulțime)

Rezultă că nicio mulțime nu se poate avea ca element, $X \notin X$,
altfel $X \ni X \ni X$... ar fi un astfel de șir

Intuitiv: orice mulțime e formată din elemente (posibil mulțimi) mai simple, care la răndul lor conțin elemente mai simple, până ajungem la elemente fundamentale

$\Rightarrow$ elimină paradoxul lui Russell

Algebra Booleană a mulțimilor

Noțiune datorată matematicianului George Boole (sec. 19)
Operațiile unei algebre Boolene (aici $\cup$ și $\cap$) satisfac legile:

Comutativitate: $A \cup B = B \cup A \qquad A \cap B = B \cap A$

Asociativitate: $(A \cup B) \cup C = A \cup (B \cup C)$ și
$(A \cap B) \cap C = A \cap (B \cap C)$

Distributivitate: $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ și
$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$

Identitate: există două valori (aici $\emptyset$ și U) astfel ca:
$A \cup \emptyset = A \qquad A \cap U = A$

Complement: orice A are un complement A^c (sau $\bar{A}$) astfel ca:
$A \cup A^c = U \qquad A \cap A^c = \emptyset$

Algebra Booleană a mulțimilor (cont.)

Alte proprietăți (pot fi deduse din cele de mai sus):

Idempotență: $A \cup A = A \qquad A \cap A = A$

Absorbție: $A \cup (A \cap B) = A \qquad A \cap (A \cup B) = A$

Dublu complement: $(A^c)^c = A$

Complemente elementelor identitate: $\emptyset^c = U \qquad U^c = \emptyset$

Limită universală: $A \cup U = U \qquad A \cap \emptyset = \emptyset$

Legile lui de Morgan:

$$(A \cup B)^c = A^c \cap B^c \qquad (A \cap B)^c = A^c \cup B^c$$

Partiție a unei mulțimi

O *partiție* a unei mulțimi A e o colecție de mulțimi $P_1, P_2, \ldots$ astfel încât:

- mulțimile $P_1, P_2, \ldots$ sunt nevide și mutual disjuncte, adică $P_i \cap P_j = \emptyset$, pentru orice $i \neq j$
- A e reuniunea tuturor mulțimilor P_i: $A = \bigcup_i P_i$

Dacă $\mathcal{A}$ e o colecție de mulțimi, definim

$$\bigcup_{A \in \mathcal{A}} A = \{x \mid x \in A_i \text{ cu } A_i \in \mathcal{A}\}$$

În particular, notăm $\bigcup_{i=1}^{n} A_i = A_1 \cup \ldots \cup A_n$ și

$\bigcup_{i \in \mathbb{N}} A_i = A_0 \cup \ldots \cup A_n \cup \ldots$ (reuniune infinită de mulțimi)

La fel pentru intersectie.

Cardinalul unei mulțimi

Cardinalul (cardinalitatea) unei mulțimi A e numărul de elemente al mulțimii. Îl notăm $|A|$.

Putem avea mulțimi *finite* sau *infinite*

Dacă A e o mulțime *finită* și $P_1, \ldots, P_N$ o partiție a ei, atunci

$$|A| = |P_1| + \ldots + |P_n|$$

Cardinalul uniunii / intersecției / diferenței

Pentru mulțimi *finite*:

Legea reuniunii:

$$|A \cup B| = |A| + |B| - |A \cap B|$$

Legea diferenței:

$$|A \setminus B| = |A| - |A \cap B|$$

Putem demonstra considerând cele 2x2 cazuri posibile:
$A \cap B$, $A \cap B^c$, $A^c \cap B$ și $A^c \cap B^c$ formează o *partiție* a universului
$A = (A \cap B) \cup (A \cap B^c)$ (partiție) $\Rightarrow |A| = |A \cap B| + |A \cap B^c|$
La fel, $\quad |B| = |A \cap B| + |A^c \cap B|$
și $\quad |A \cup B| = |A \cap B| + |A \cap B^c| + |A^c \cap B|$
de unde, combinând, rezultă egalitățile de mai sus.

Principiul includerii și excluderii

pentru mulțimi *finite*

$$|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C|$$

Mai general,

$$|\bigcup_{i=1}^{n} A_i| = \sum_{i=1}^{n} |A_i| - \sum_{1 \leq i < j \leq n} |A_i \cap A_j| + \ldots + (-1)^n |A_1 \cap \ldots A_n|$$

Demonstrație: prin inducție după n

Tupluri și produsul cartezian

Un *n-tuplu* e un șir de n elemente $(x_1, x_2, \ldots, x_n)$
(nu neapărat distincte, iar ordinea elementelor contează).

Cazuri particulare: *pereche* (a, b), *triplet* (x, y, z), etc.

Produsul cartezian a două mulțimi e mulțimea perechilor

$$A \times B = \{(a, b) \mid a \in A, b \in B\}$$

Exemplu: mulțimea numerelor complexe poate fi văzută ca produs cartezian $\mathbb{R} \times \mathbb{R}$ (putem găsi o *bijecție* între ele)

Produsul cartezian a n mulțimi e mulțimea $n - tuplelor$

$$A_1 \times A_2 \times \ldots \times A_n = \{(x_1, x_2, \ldots, x_n) \mid x_i \in A_i, 1 \leq i \leq n\}$$

Dacă mulțimile sunt finite, atunci

$$|A_1 \times A_2 \times \ldots \times A_n| = |A_1| \cdot \ldots \cdot |A_n|$$

Mulțimea submulțimilor

Mulțimea submulțimilor (engl. *power set*) a unei mulțimi S, notată $\mathcal{P}(S)$ (uneori 2^S):

$$\mathcal{P}(S) = \{X \mid X \subseteq S\}$$

Exemplu, pentru $S = \{a, b, c\}$, avem
$\mathcal{P}(S) = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\}\}$

Dacă S e finită, atunci $|\mathcal{P}(S)| = 2^{|S|}$

Există o bijecție între $\mathcal{P}(S)$ și mulțimea funcțiilor $f : S \to \{0, 1\}$
- dacă $f(x) = 1$, x aparține submulțimii, altfel nu.
- numărul funcțiilor e $|\{0, 1\}|^{|S|} = 2^{|S|}$

Mulțimi numărabile și nenumărabile

Informal: o mulțime e numărabilă dacă putem da fiecărui element un număr (natural, diferit).

Altfel spus: O mulțime e *numărabilă* dacă are cardinalul egal cu cardinalul unei submulțimi a numerelor naturale. Sau, formal:

O mulțime S e *numărabilă* dacă există o funcție injectivă $f : S \to \mathbb{N}$

Orice mulțime finită e numărabilă: $|A| = n \Rightarrow A = \{a_1, a_2, ...a_n\}$ (indicii reprezintă corespondența cu $\{1, 2, ...n\}$)

Dar nu orice mulțime numărabilă e finită

$\mathbb{N}$ e numărabilă: în definiție, luăm f funcția identitate

$\mathbb{Z}$ e numărabilă: putem enumera: $0, -1, 1, -2, 2, ...$
$f(x) = 2x$, pentru $x \geq 0$, $f(x) = -2x - 1$ pentru $x < 0$

Definiție echivalentă: S e numărabilă dacă e fie finită, fie există o bijecție între S și $\mathbb{N}$ (e *infinit numărabilă*).

Numerele raționale sunt numărabile

1/1	1/2	1/3	1/4	...
2/1	2/2	2/3	2/4	...
3/1	3/2	3/3	3/4	...
...	...	...	...	...

NU putem ~~număra elementele pe linii~~: deja prima linie e *infinită*, nu ajungem niciodată la a doua!

Enumerăm pe *diagonale*
(după valoare crescătoare a lui $m + n$, numărător + numitor):
$1/1, \quad 1/2, 2/1, \quad 1/3, 2/2, 3/3, \quad 1/4, 2/3, 3/2, 4/1, \ \ldots$
$\Rightarrow$ orice element va fi numărat

Exercițiu: care e numărul de ordine al lui m/n ?

Tehnică generală:
asociem fiecărui element o *mărime*: aici m+n; lungimea la șiruri; etc
așa încât cu fiecare mărime să avem un număr *finit* de elemente
numărăm după mărime crescătoare $\Rightarrow$ ajungem la fiecare element

Construcții cu mulțimi numărabile

O mulțime e numărabilă dacă putem enumera elementele într-un șir:
Un șir e o funcție de la $\mathbb{N}$ la mulțimea elementelor șirului
(sau de la $\{1, 2, \dots n\}$ la elementele șirului, pentru un șir finit)
Reuniunea a două mulțimi numărabile e numărabilă.

Enumerăm *alternativ* mulțimile (similar cu cazul lui $\mathbb{Z}$):
$A = \{a_1, a_2, \dots a_n, \dots\}$, $B = \{b_1, \dots, b_n, \dots\}$ (le putem enumera)
$\Rightarrow$ formăm șirul $a_1, b_1, a_2, b_2, \dots, a_n, b_n, \dots$
(putem avea duplicate, oricum am enumerat toate elementele)

Produsul cartezian
Produsul cartezian $A \times B$ a două mulțimi numărabile e numărabil
Folosim aceeași construcție ca la numerele raționale:
enumerăm perechile în ordine crescătoare a sumei indicilor:
$A \times B = \{(a_1, b_1), (a_1, b_2), (a_2, b_1), (a_1, b_3), (a_2, b_2), (a_3, b_1), \dots\}$

Prin inducție, pentru reuniunea / produsul cartezian a *n* mulțimi

Realii sunt nenumărabili

construcția diagonală a lui Cantor:

Reprezentăm numerele subunitare în baza 2: cifrele sunt 0 și 1

Exemplu:

$0.01101... = 0 + 0 \cdot 2^{-1} + 1 \cdot 2^{-2} + 1 \cdot 2^{-3} + 1 \cdot 2^{-4} + 1 \cdot 2^{-5} + \ldots$

Presupunem prin reducere la absurd că realii din [0, 1) ar fi numărabili

$\Rightarrow$ am putea scrie realii subunitari într-un tabel, după numărul de ordine

$$\begin{array}{llllll} r_1 = 0. & d_{11} & d_{12} & d_{13} & \ldots & 0.1011101... \\ r_2 = 0. & d_{21} & d_{22} & d_{23} & \ldots & 0.0110010... \\ r_3 = 0. & d_{31} & d_{32} & d_{33} & \ldots & 0.1101101... \\ \ldots & \ldots & \ldots & & & \end{array}$$

Construim un număr real $\mathbf{x} = 0.d_1 d_2 d_3 \ldots$ cu următoarele cifre:

$d_i = 1 - d_{ii}$ (urmărind diagonala matricii, schimbăm $0 \leftrightarrow 1$)

Dar $\mathbf{x}$ diferă de toate numerele din tabel (diferă de r_i la poziția i)!

Deci *mulțimea realilor $\mathbb{R}$ e nenumărabilă !*

Există oricâte infinituri

Teorema lui Cantor: *Nu există bijecție* de la X la $\mathcal{P}(X)$.

Să presupunem că ar exista o bijecție $f : X \to \mathcal{P}(X)$.
Formăm mulțimea:

$$Y = \{x \in X \mid x \notin f(x)\}$$

Cum $Y \in \mathcal{P}(X)$, și f e bijecție, există $y \in X$ cu $f(y) = Y$.

Dacă $y \in Y$, cum $Y = f(y)$ atunci $y \in f(y)$, și nu respectă condiția de construcție a lui Y, deci $y \notin Y$, contradicție.
Dacă $y \notin Y$, atunci $y \notin f(y)$ și satisface condiția pentru Y, deci $y \in Y$, contradicție.

Deci presupunerea e falsă, nu poate exista o bijecție.
Construcția seamănă cu cea din paradoxul lui Russell,
dar cu alt scop: demonstrația prin *reducere la absurd*.

Deci $|\mathbb{N}|$, $|\mathcal{P}(\mathbb{N})|$, $|\mathcal{P}(\mathcal{P}(\mathbb{N}))|$, ... sunt infinități tot mai mari, incomparabile!

Calculabilitate și problema terminării (halting problem)

pusă de Alan Turing pentru *mașina (automatul) lui Turing*, un model simplu, universal de calcul. În formularea pentru programe:

Nu există algoritm (program) care ia un program arbitrar P și un set de date D și determină dacă $P(D)$ (rularea lui P cu datele D) s-ar termina (opri) sau ar rula la infinit.

Presupunem că ar exista un astfel de program *CheckHalt(P, D)*.
Deci, *CheckHalt*(X, X) spune ce face prog. X cu textul său ca date

Construim un "program imposibil" care face opusul a ceea ce face!

Întâi, definim programul *Test(X)* având ca intrare un program X:
- dacă *CheckHalt*(X, X) decide `"halt"`, atunci **ciclează la infinit**
- dacă *CheckHalt*(X, X) decide `"ciclează"`, atunci **stop**

Deci *CheckHalt*(X, X) spune ce face X(X) iar *Test*(X) face opusul

Se oprește *Test*(Test)? Răspunsul e dat de *CheckHalt*(Test,Test).
dar *Test*(Test) (cu X=*Test*) face *opusul* lui *CheckHalt*(Test,Test)
$\Rightarrow$ *contradicție*, deci nu poate exista *CheckHalt*!

Mulțimi în ML: modulul Set

Întâi instanțiem un *modul* pentru lucru cu mulțimi:

```
module S = Set.Make(String) (* String: modul de lucru cu siruri*
(* S = modul (tipuri+functii) pentru multimi de siruri *)
(* functii standard: t = tip multime, elt = tip element *)
(* val mem : elt -> t -> bool
   val cardinal : t -> int
   val elements : t -> elt list *)

let s1 = S.add "ana" (S.add "bob" (S.singleton "cora"))
```

nu există sintaxă specială ~~{ "ana", "bob", "cora" }~~

OCaml necesită o funcție de *comparare* pe elementele unei mulțimi.
⇒ un *modul* care definește tipul element și funcția de comparare

```
module Int = struct
  type t = int
  let compare = compare
end
(* Char, String: similare, dar predefinite *)
```

Parcurgerea mulțimilor

Mulțimile nu au un element special (v. capul listei)
deși `choose : t -> elt` ne dă un element (oarecare)
⇒ e important să folosim funcțiile de parcurgere

```
val iter : (elt -> unit) -> t -> unit
val fold : (elt -> 'a -> 'a) -> t -> 'a -> 'a
```

ordinea parametrilor la `fold` e ca la `List.fold_right`
Putem defini de exemplu `union` folosind `fold`

```
let union s1 s2 = S.fold (fun e s-> S.add e s) s1 s2
(* parcurge s1, adauga fiecare element, pornind de la s2 *)
let union = S.fold S.add
```