Kyuubi

[Proiect SBC]

Nichita Mihaela, 1404B

Antonache Andreea, 1403A

Cuprins

[*Abstract* 3](#_Toc5751204)

[ETAPA II 3](#_Toc5751205)

[*1.* *Enuntul problemei. Obiective* 3](#_Toc5751206)

[*2.* *Descrierea universului de discurs* 3](#_Toc5751207)

[*3.* *Scenarii considerate* 4](#_Toc5751208)

[*4.* *Strategii de rezolvare* 7](#_Toc5751209)

[*5.* *Definirea bazei de fapte* 9](#_Toc5751210)

[ETAPA III 11](#_Toc5751211)

[*6.* *Definirea bazei de reguli* 11](#_Toc5751212)

[*7.* *Rezolvarea conflictelor* 12](#_Toc5751213)

[*7.1* *Cazuri conflictuale* 12](#_Toc5751214)

[*7.2* *Strategii de rezolvare conflicte* 13](#_Toc5751215)

[*8.* *Schema logica de functionare* 14](#_Toc5751216)

[ETAPA IV 16](#_Toc5751217)

[*9.* *Utilizare Clips in alta aplicatie* 16](#_Toc5751218)

[*Concluzii* 18](#_Toc5751219)

[*Rezumat* 18](#_Toc5751220)

[*Bibliografie* 18](#_Toc5751221)

## *Abstract*

*Kyuubi* este o aplicatie care surprinde un sistem expert dezvoltat in mediul CLIPS, un limbaj de programare bazat pe reguli. Acest sistem bazat pe reguli rezolva cubul rubik pornind de la o configuratie initiala data de catre utilizator din interfata GUI, actiune facilitata de vizualizarea cubului rubik. Complexitatea algoritmului pe care si omul il poate aborda, este cu atat mai dificila cu cat apar conflicte pentru sistem, conflicte pe care omul le poate solutiona foarte usor, dar pentru sistem este mai dificil.

# ETAPA II

## *Enuntul problemei. Obiective*

Obiectivul dezvoltării sistemului expert de tip planificator este rezolvarea cubului Rubik, pornind de la configuratie initiala in care fata alba este rezolvata.

Ipoteze simplificatoare:

* Orientarea fețelor alb și galben rămâne neschimbată: Down = White, Up = Yellow
* Orientarea celorlalte fețe se poate schimba, pornind însă de la starea inițiala dată:

Front = Red, Right = Green, Back = Orange, Left = Blue,

* Fața corespunzătoare culorii alb (White) și orientării Down este rezolvată

## *Descrierea universului de discurs*

Mediul de dezvoltare a sistemului expert este realizat în Clips. Conceptele de interes vizează un algoritm de rezolvarea cubului rubik, un algoritm care poate fi realizat atât de om, cât și de calculator pe baza unor reguli, relații și proprietăți ale obiectului în cauză, cubul rubik.

A. Concepte

* Față-Culoare
* Pătrat-Culoare
* Orientare
* Strat

B. Proprietăți

* Stratul mijlociu este rezolvat
* Fața galbenă are cruce galbena
* Fața galbenă are o linie galbena
* Fața albă este rezolvată
* Mulțimea de pătrate formează o față

C. Relații

* Relații între conceptele Față-Culoare și Orientare:

Exemplu: Fața corespunzătoare culorii roșu are orientarea F (front)

* Relații între conceptele Față-Culoare și mai multe instanțe ale conceptului Culoare-Pătrat:

Exemplu: Fața corespunzătoare culorii verde (G) conține culorile: Y G Y Y G B G G G

D. Acțiuni

* Acțiunea de rotire a feței frontale (F) în sensul acelor de ceas
* Acțiunea de rotire a feței frontale (Fi) în sensul opus acelor de ceas
* Acțiunea de rotire a feței din spate (B) în sensul acelor de ceas
* Acțiunea de rotire a feței din spate (Bi) în sensul opus acelor de ceas
* Acțiunea de rotire a feței din dreapta (R) în sensul acelor de ceas
* Acțiunea de rotire a feței din dreapta (Ri) în sensul opus acelor de ceas
* Acțiunea de rotire a feței din stânga (L) în sensul acelor de ceas
* Acțiunea de rotire a feței din stânga (Li) în sensul opus acelor de ceas
* Acțiunea de rotire a feței de dedesubt (D) în sensul acelor de ceas
* Acțiunea de rotire a feței de dedesubt (Di) în sensul opus acelor de ceas
* Acțiunea de rotire a feței de deasupra (U) în sensul acelor de ceas
* Acțiunea de rotire a feței de deasupra (Ui) în sensul opus acelor de ceas
* Acțiunea de crearea a secvenței de rotații necesare pentru atingerea unui obiectiv, în funcție de starea curentă
* Acțiunea de reorientare a fețelor laterale în sensul executării corecte a rotațiilor, ceea ce presupune redefinirea relațiilor dintre culorile fetelor (R, G, O, B) și orientările acestora (F, R, B, L)

Notă: Dacă, în cele ce urează, o secvență de acțiuni cere efectuarea unei acțiuni notată cu X2, X aparținând mulțimii {F, B, R, L, U, D}, atunci se execută două rotiri consecutive a feței X.

## *Scenarii considerate*

1. Instanța 1

* stare inițială: Fața corespunzătoare culorii alb este rezolvată



* obiectiv: Rezolvarea stratului mijlociu



1. Instanța 2

* stare inițială: Stratul mijlociu este rezolvat
* obiectiv: Formarea unei cruci galbene pe fața de sus



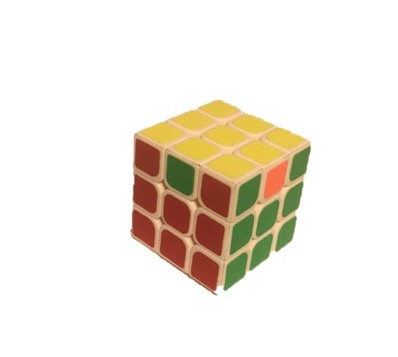
1. Instanța 3

* stare inițială: Existența unei cruci galbene pe fața de sus
* obiectiv: Rezolvarea feței de sus (de culoare galbenă)



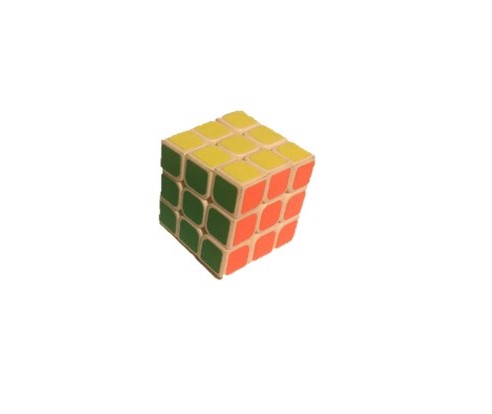
1. Instanța 4

* stare inițială: Fața corespunzătoare culorii galben este rezolvată
* obiectiv: Rezolvarea colțurilor din stratul superior



1. Instanța 5

* stare inițială: Colțurile din stratul superior sunt rezolvate
* obiectiv: Rezolvarea marginilor din stratul superior (cub rezolvat)



## *Strategii de rezolvare*

1. Pentru *instanța 1* - Rezolvarea stratului mijlociu:
2. Dacă nu există nici un mini-cub de tip margine în stratul superior care să aibă ambele fețe diferite de galben, atunci se execută următoarele acțiuni:

* Se reorientează fețele asa încât fața laterală, pentru care pătratul din mijloc și cel din dreapta lui au culori diferite, să devină fața frontală (F)
* U, R, Ui, Ri, Ui, Fi, U, F

1. Dacă există un mini-cub de tip margine în stratul superior care să aibă ambele fețe diferite de galben, atunci se execută următoarele acțiuni:

* Dacă culoarea feței laterale a mini-cubului de tip margine este diferită de culoarea pătratului din centrul feței respective, atunci se rotește fața de sus (U) până când acestea devin identice
* Se reorientează fețele așa încât fața laterală în componenta căreia intra mini-cubul de interes să fie cea frontală (F)
* Dacă un mini-cub de tip margine din stratul superior trebuie să ajungă pe fața din dreapta (R), atunci se execută următoarele acțiuni: U, R, Ui, Ri, Ui, Fi, U, F
* Dacă un mini-cub de tip margine din stratul superior trebuie să ajungă pe fața din stânga (L), atunci se execută următoarele acțiuni: Ui, Li, U, L, U, F, Ui, Fi

1. Pentru *instanța 2* - Formarea unei cruci galbene pe fața de sus:
2. Dacă există o linie formată din 3 pătrate galbene pe fața de sus (U), atunci se execută următoarele acțiuni:

* Se reorientează fetele așa încât fața frontală (F) să fie paralelă cu linia respectivă
* F, R, U, Ri, Ui, Fi

1. Dacă nu există o linie formată din 3 pătrate galbene pe fața de sus (U), atunci se execută următoarele acțiuni:

* Dacă pe fața de sus (U) se găsesc 3 pătrate de culoare galbenă care să formeze litera L, atunci se rotește fața de sus (U) până când se ajunge la starea în care cuburile de tip margine corespunzătoare capetelor literei L să aparțină și fețelor din stânga (L), respectiv, spate (B)
* F, U, R, Ui, Ri, Fi

1. Pentru *instanța 3* - Rezolvarea feței de sus (de culoare galbenă) - Algoritmul “Sune”:
2. Dacă pe fața de sus (U) nu există nici un colț de culoare galbenă, atunci se execută următoarele acțiuni:

* Se reorientează fețele laterale așa încât pătratul din colțul din dreapta-sus al feței din stânga (L) să fie de culoare galbenă
* SEQ1 = R, U, Ri, U, R, U2, Ri

1. Dacă pe fața de sus (U) există 2 colțuri de culoare galbenă, atunci se execută următoarele acțiuni:

* Se reorientează fețele laterale așa încât pătratul din colțul din stânga-sus al feței frontale (F) să fie de culoare galbenă
* SEQ1

1. Dacă pe fața de sus (U) există exact un colț de culoare galbenă, atunci se execută următoarele acțiuni:

* Se reorientează fețele laterale așa încât mini-cubul corespunzător colțului respectiv să aparțină și fețelor din stânga (L) și din fața (F)
* Dacă pătratul din dreapta-sus de pe fața frontală (F) este de culoare galbenă, atunci se execută secvența de acțiuni SEQ1
* Dacă pătratul din stânga-sus de pe fața din dreapta (R) este de culoare galbenă, atunci se execută următoarea secvență de acțiuni: U2, R, U2, Ri, Ui, R, Ui, Ri

1. Pentru *instanța 4* - Rezolvarea colțurilor din stratul superior:
2. Dacă nu există două colțuri consecutive orientate corect (un colț este orientat corect dacă toate culorile fețelor acestuia corespund culorii feței pe care se află), atunci se rotește fața de sus (U)
3. Dacă, după 3 rotații ale feței de sus, nu se găsesc două colțuri consecutive orientate corect, atunci se execută următoarea secvență de acțiuni:

SEQ2 = Ri, F, Ri, B2, R, Fi, Ri, B2, R2,Ui

1. Dacă există două colțuri consecutive orientate corect, atunci se execută următoarele acțiuni:

* Se reorientează fețele laterale așa încât cele două colțuri să se afle pe fața din spate (B)
* SEQ2

1. Pentru *instanța 5* - Rezolvarea marginilor din stratul superior
2. Dacă niciuna dintre fețele laterale nu este rezolvată, atunci:

* Se execută secvența de acțiuni SEQ3 = F2, U, L, Ri, F2, Li, R, U, F2

1. Dacă una dintre fețele laterale este rezolvata, atunci:

* Se reorientează fețele laterale așa încât fața rezolvată să devină fața din spate (B)
* Dacă culoarea cubului de tip margine dintre fața frontală și cea de sus corespunde culorii feței din dreapta (sensul acelor de ceas), atunci se execută SEQ3
* Dacă culoarea cubului de tip margine dintre fața frontală și cea de sus corespunde culorii feței din stânga (sens trigonometric), atunci se execută următoarea secvență de acțiuni:

F2, Ui, L, Ri, F2, Li, R, Ui, F2

## *Definirea bazei de fapte*

a. Tipare de fapte

(face-orientation <face-color> <orientation>)

(face <<<square-color>>>)

(is-done <face-color>)

(is-done <layer>)

(has-line <face-color>)

(has-cross <face-color>)

(move <priority> <rotation>)

(Solution <<<rotatie/re-orientare>>>)

b. Descriere prin fapte nestructurate

Baza de Fapte pentru instanța 1 din secțiunea 3 arată în felul următor:

* Stare inițială: Fața corespunzătoare culorii alb este rezolvată. Cubul arată ca în Figura 5.1

(face W W W W W W W W W)

(face Y Y R R Y Y Y O G)

(face G B O R R R R R R)

(face Y G Y G G Y G G G)

(face B B O O O O O O O)

(face B Y R G B B B B B)

(face-orientation R F)

(face-orientation G R)

(face-orientation O B)

(face-orientation B L)



* După atingerea obiectivului: Stratului mijlociu rezolvat

(face W W W W W W W W W)

(face Y R G G G G G G G)

(face B Y O R R R R R R)

(face Y R Y G G G G G G)

(face R G B O O O O O O)

(face R Y O B B B B B B)

(face-orientation G F)

(face-orientation O R)

(face-orientation B B)

(face-orientation R L)

(is-done middle-layer)



# ETAPA III

## *Definirea bazei de reguli*

● Acțiunea de rotire a feței frontale (F) în sensul acelor de ceas:

Daca exista in BF scopul dat prin faptul (move <prioritate> F/Fi/U/Ui/R/Ri/L/Li/B/Bi/D/Di) si nu exista un alt fapt ce respecta acelasi tipar, dar cu prioritate mai mica, atunci se actualizeaza configuratia fetelor cubului si se sterge scopul care a activat regula.

Cazurile in care actiunea nu este aplicabila:

o In BF nu exista vreun fapt de forma (move <prioritate> <rotatie>)

o In BF exista un fapt de forma (move <prioritate> <rotatie>) si un alt fapt de aceeasi forma, dar cu prioritate mai mica

● Acțiunea de reorientare a fețelor laterale în sensul executării corecte a rotațiilor, ceea ce presupune redefinirea relațiilor dintre culorile fetelor (R, G, O, B) și orientările acestora (F, R, B, L)

Daca in BF exista scopul dat prin faptul (must-be-front <color>) si culoarea indicata de acesta este orientata spre dreapta(R), atunci se actualizeaza configuratiile fetelor U si D, precum si orientarile fetelor laterale, imitand o rotatie de 90 de grade a cubului in jurul axului sau in sensul acelor de ceas.

Daca in BF exista scopul dat prin faptul (must-be-front <color>) si culoarea indicata de acesta este orientata spre stanga(L), atunci se actualizeaza configuratiile fetelor U si D, precum si orientarile fetelor laterale, imitand o rotatie de 90 de grade a cubului in jurul axului sau in sens invers acelor de ceas.

Daca in BF exista scopul dat prin faptul (must-be-front <color>) si culoarea indicata de acesta este orientata spre spate(B), atunci se actualizeaza configuratiile fetelor U si D, precum si orientarile fetelor laterale, imitand o rotatie de 180 de grade a cubului in jurul axului sau.

Cazul in care actiunea nu este aplicabila:

* In BF nu exista vreun fapt de forma (must-be-front <color>)

● Acțiunea de crearea a secvenței de rotații necesare pentru atingerea unui obiectiv, în funcție de starea curentă

Daca stratul mijlociu nu este rezolvat si toate mini-cuburile de tip margine ce apartin fetei de sus (U) au una din fete de culoare galben, atunci se adauga in BF faptele de forma: (must-be-front <color>), (move <prioritate> <rotatie>), reprezentand scopurile necesare obtinerii unui mini-cub de tip margine pe fata frontala care sa nu aiba nici o fata galbena.

Daca stratul mijlociu nu este rezolvat si o fata laterala a cubului are cel de-al doilea patrat (#2) de aceeasi culoare cu cel de-al 5-lea, iar patratul ce corespunde patratului #2 pe fata de sus (X) este de aceeasi culoare cu fata din dreapta/stanga a cubului (R/L), atunci se adauga in BF faptele de forma: (move <prioritate> <rotatie>), reprezentand scopurile necesare deplasarii patratului (X) pe fata din dreapta/stanga.

Dupa rularea repetata a acestor reguli, se atinge obiectivul rezolvarii stratului mijlociu.

Daca stratul mijlociu este rezolvat, nu s-a obtinut inca o cruce galbena pe fata de sus si exista o linie galbena pe fata de sus, pozitionata pe mijloc, atunci se adauga in BF faptele de forma:(must-be-front <color>), (move <prioritate> <rotatie>), reprezentand miscarile necesare obtinerii acelei cruci galbene.

Daca s-a obtinut o cruce galbena pe fata de sus si nici unul dintre colturile fetei de sus nu este de culoare galbena, iar cel de-al treilea patrat de pe fata frontala este de culoare galbena, atunci se adauga in BF faptele de forma: (move <prioritate> <rotatie>), reprezentand miscarile necesare obtinerii cel putin a unui colt galben pe fata de sus.

Daca s-a obtinut o cruce galbena pe fata de sus si exista unul sau mai multe colturi de culoare galbena pe fata de sus, atunci se adauga in BF faptele de forma: (move <prioritate> <rotatie>), reprezentand miscarile necesare rezolvarii fetei de sus.

## *Rezolvarea conflictelor*

### *Cazuri conflictuale*

Se considera instanta 1 a problemei. La nivelul acesteia apar urmatoarele conflicte date prin schimbarea strategiei de lucru a motorului de inferență:

1. Daca la un moment dat in BF exista atat fapte ce activeaza regulile pentru rotatii, cat si una din regulile de reorientare a fetelor cubului (change-orientation-left-to-front, change-orientation-right-to-front, change-orientation-back-to-front), atunci apare un conflict intre cele doua categorii de reguli.

2. Dupa ce se adauga in BF o secventa de fapte reprezantand rotatiile care trebuie efectuate, se vor activa simultan atat regulile care materializeaza acele rotatii (move-R,move-Ri, move-U,move-Ui, move-F,move-Fi,move-L,move-Li,move-B,move-Bi,move-D,move-Di), cat si alte reguli specifice starii curente a cubului (din instanta 1).

3. In momentul in care stratul mijlociu indeplineste conditiile pentru a fi considerat rezolvat, se activeaza simultan mai multe reguli: regula R1-middle-layer-done, care introduce in BF faptul (is-done middle-layer) si regula R1a-all-edge-pieces-contain-yellow, care trateaza cazul in care toate mini-cuburile de tip margine ce apartin fetei de sus (U) au una din fete de culoare galben.

4. Se pot activa simultan reguli specifice unor instante diferite, de exemplu: o regula din instanta 1 (etapa a rezolvarii cubului in care stratul mijlociu nu este rezolvat) si o regula din instanta 2 (etapa in care stratul mijlociu este rezolvat).

### *Strategii de rezolvare conflicte*

1. Pentru evitarea conflictului dintre o *regula de tip rotatie* si *una de tip reorientare*, dat fiind faptul ca reorientarea trebuie sa fie efectuata inaintea ratatiilor, prioritatea cea mai mare ii revine regulii de reorientare.
2. Pentru evitarea conflictului dintre *regulile ce efectueaza rotatiile* si *alte reguli*, se aplica o prioritate mai mare celor 12 reguli reprezentand rotatiile.
3. Pentru evitarea conflictului dintre regula *R1-middle-layer-done* si *alte reguli*, acesteia i se aplica o prioritate mai mare decat a tuturor celorlalte reguli specifice instantei 1.
4. Pentru evitarea conflictelor dintre reguli specifice unor *instante(etape) diferite*, se foloseste mecanismul de excludere mutuala. Pentru r*egulile specifice unei instante*(ex. instanta 2) se adauga in partea stanga a tuturor regulilor tiparul ce presupune obiectivul negat (not (has-cross Y)) si cel ce comunica obiectivul instantei anterioare atins (is-done middle-layer).

*Same color prioritatea mai mare decat different color – instanta 1.*

## *Schema logica de functionare*

A close up of a map

Description automatically generated

*Fig.* 1 – *Schema logica Kyuubi – fata galbena rezolvata*

Sunt atasate cele 3 instante realizate manual pentru o mai buna exemplificare si intelegere a structurii logice.

A picture containing whiteboard, text

Description automatically generated

*Fig.* 2 – *Instanta 1*

A close up of text on a whiteboard

Description automatically generated

*Fig.* 3 – *Instanta 2*

A picture containing text, whiteboard

Description automatically generated

*Fig.* 4 – *Instanta 3*

# ETAPA IV

## *Utilizare Clips in alta aplicatie*

Mini-aplicatia de tip UI este dezvoltata cu ajutorul limbajului de programare Python. Aceasta permite incarcarea si rularea programului Clips (“*rubiks-cube.clp*”). Utilizatorul are posibilitatea de a introduce configuratia cubului rubik prin intermediul reprezentarii desfasurate a acestuia (fetele *Blue*, *Red*, *Green*, *Orange*, *Yellow* si *White*). Initial, cubul desfasurat are toate fetele rezolvate.

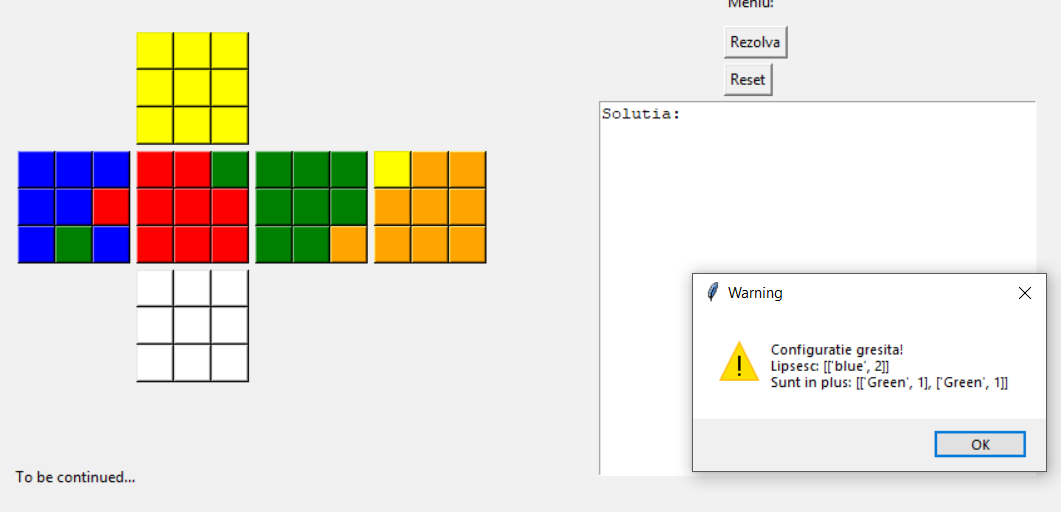
A screenshot of a cell phone

Description automatically generated

*Fig.* 5 – *Kyuubi GUI*

Meniul consta din 2 butoane : *Rezolve* si *Reset*.

Utilizatorul, pe fiecare mini-patrat, plaseaza cursorul, click dreapta pana in momentul in care acesta devine *culoarea* dorita de utilizator pentru configurarea propriei stari, mai putin modificarea fetei *White*. Pentru rularea programului, utilizatorul actioneaza butonul “*Rezolve*”.



*Fig.* 6 – *Configuratie cub incorecta*

In cazul in care configuratia este gresita (s-au setat mai multe/putine patratele de o culoare decat numarul necesar), utilizatorul primeste avertisment, precum in imaginea de mai jos :

A screenshot of a cell phone

Description automatically generated

*Fig.* 7 – *Solutia*

*Fig.* 7 – *Solutia corecta*

In fereastra cu textul “*Solutia:*” este afisat un *fapt cu solutia.*

## *Concluzii*

Rezolvarea cubului rubik este realizata prin intermediul unor reguli, care pe baza unei configuratii initiale a cubului, actioneaza daca sunt respectate anumite fapte. ***Cubul rubik rezolvat*** este rezultatul intregului sistem bazat pe reguli ce opereaza cu starea initiala a cubului.

Un aspect *negativ* este faptul ca pornim de la o configuratie in care fata alba este rezolvata. O imbunatatire este rezolvarea efectiva a cubului rubik pornind de la o configuratie in care nu exista nici o fata rezolvata.

## *Rezumat*

Aplicatia “*Kyuubi*”dezvoltata in mediul de dezvoltare CLIPS, este un sistem expert care rezolva o fata a cubului rubik, pornind de la o configuratie initiala unde fata alba este rezolvata. Algoritmul sustine logica sistemului expert intr-o maniera pe care si omul o poate aborda, un algoritm implementat pe baza de reguli si prioritati ce definesc o armonie frumoasa si rezolva conflictele aparute pe parcursul aplicatiei.

## *Bibliografie*

[1] Panescu D., Pascal C., Programare bazată pe reguli, Îndrumar de laborator, Editura Conspress, București, 2013, ISBN 978-973-100-258-3.