

## Capitolul 2: Noțiuni introductive despre Programul Rodin

“Limbaajul **Rodin**, este un limbaj pseudocod nou, implementat în 2009, în stil C, proiectat pentru profesorii și elevii din România, pentru orele de algoritmistică. Acest program a fost realizat în întregime de Lect. univ. drd. Dan Popa.” Așa este prezentat acest limbaj pe prima pagină a site-ului său <http://www.haskell.org/haskellwiki/Rodin>.

**Rodin** poate fi descărcat gratuit de pe <http://www.haskell.org/haskellwiki/Rodin/Download>. Tot în cadrul acestui site puteți găsi și noutăți referitoare la acest program, cu mai multe exemple realizate de studenții de la specializările Matematică și Informatică de la Universitatea „Vasile Alecsandri” din Bacău.

Programul funcționează atât pe sistemul de operare LINUX cât și pe WINDOWS. Versiunea prezentată puțin mai jos este pentru sistemul de operare WINDOWS.

Deși limbajele de programare permit de obicei să împărțim aplicația în programul principal și alte subprograme, în Rodin nu sunt implementate subprograme, deocamdată. Tot fișierul text conține deci programul principal, urmat eventual de un text format din comentarii. Pentru a asocia instrucțiunile programului unei anumite secțiuni, instrucțiunile trebuie plasate între două acolade ("{" "}"). Pentru fiecare acoladă deschisă, trebuie să nu uitați perechea ei, care închide grupul de instrucțiuni. Uneori este necesar ca programul să execute sau nu, un set de instrucțiuni *daca* este adevărat (sau fals) rezultatul unei condiții, dar și în acest caz sunt prezente ambele acolade care marchează setul, precum și cuvinte ca: *atunci*, *altfel*.

Pe măsură ce programele dumneavoastră devin mai complexe, vor apărea situații în care un program va *repetă* același set de instrucțiuni de un anumit număr de ori sau *până când* se îndeplinește o anumită condiție sau va *executa* un grup de instrucțiuni *cât timp* se menține o anumită condiție.

Atunci când programele repetă una sau mai multe instrucțiuni până când este îndeplinită o condiție dată, se spune că programul execută o procesare iterativă. O trecere a programului prin instrucțiunile care se repetă se numește iterație. Când lucrați cu iterative și condiționale, cuvintele cheie scrise italic (înclinat) mai sus le putem folosi la scrierea programului în limbajul **Rodin**.

După ce ați descărcat programul, îl dezarhivați unde doriți, de exemplu pe Desktop, utilizând arhivatorul din Windows, WinRAR, 7ZIP sau orice altă aplicație de dezarhivare doriți. Apoi intrați în START -> RUN, scrieți **cmd** și apăsați tasta <ENTER>.

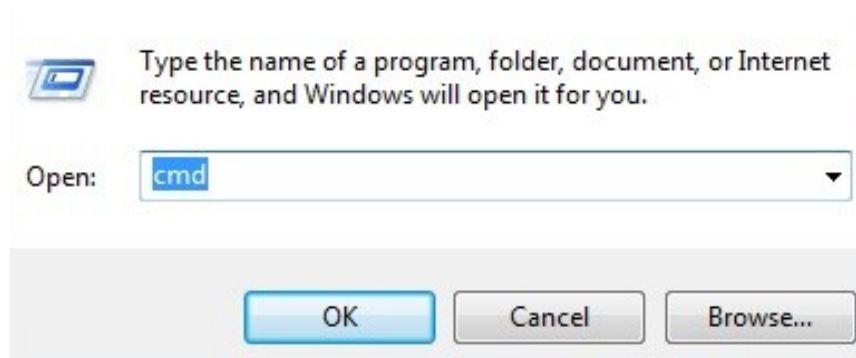


Fig.10 (deschiderea ferestrei terminal)

După ce ați apăsăat <OK> trebuie să arate ca în imaginea de mai jos, cu excepția numelui utilizatorului.

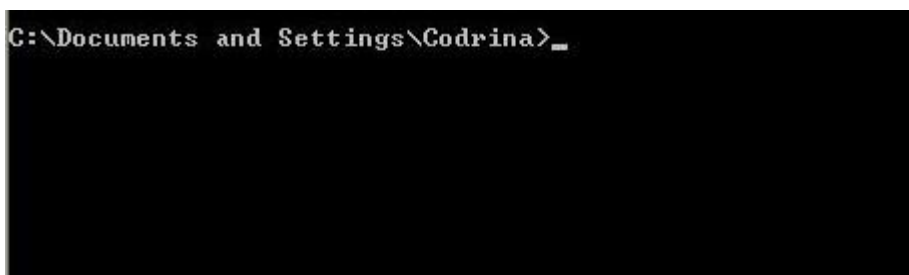


Fig.11 (primul pas către programul executabil)

Pe ecran va apărea o fereastră de comenzi, în acea casetă scrieți “cd desktop” ca în figura următoare și apoi apăsați <ENTER>.



Fig.12 (intrarea în folder-ul DESKTOP)

Apoi scrieți “cd experimentexp12win” (“cd experimentexp12win” fiind folder-ul unde am salvat programul) și apăsați <ENTER>. Ar trebui să arate exact ca în imaginea de mai jos.



```
C:\Documents and Settings\Codrina\Desktop>cd experimentexp12win_
```

Fig.13 (intrarea în folder-ul ExperimentExp12Win)

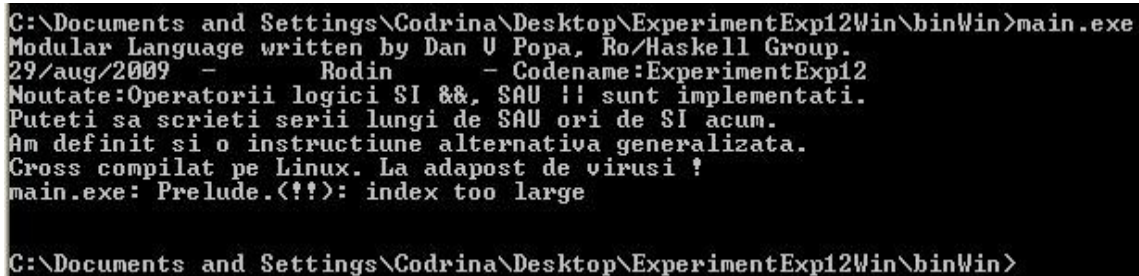
Încă o dată vom apela la comanda "cd" (change directory) pentru a ajunge la fișierul rădăcină al programului Rodin. Folosind comanda "cd binwin" și apăsând <ENTER>, trebuie să ajungeti la ceva ca în imaginea de mai jos:



```
C:\Documents and Settings\Codrina\Desktop\ExperimentExp12Win>cd binwin_
```

Fig.14 (intrarea în folder-ul binwin)

Pentru a rula un exemplu folosim executabilul "main.exe". Scrieti "main.exe" și apăsați <ENTER>, ar trebui să arate ca în imaginea de mai jos. Mesajul de eroare este normal, Rodin negasind un program exemplu de executat. În mod normal după main.exe se scrie numele acestui program, cu extensie .txt.



```
C:\Documents and Settings\Codrina\Desktop\ExperimentExp12Win\binWin>main.exe
Modular Language written by Dan U Popa, Ro/Haskell Group.
29/aug/2009 - Rodin - Codename:ExperimentExp12
Noutate:Operatorii logici SI &&, SAU || sunt implementati.
Puteti sa scrieti serii lungi de SAU ori de SI acum.
Am definit si o instructiune alternativa generalizata.
Cross compilat pe Linux. La adapost de virusi !
main.exe: Prelude.<!!>: index too large

C:\Documents and Settings\Codrina\Desktop\ExperimentExp12Win\binWin>
```

Fig.15 (verificarea executabilului)

Urmărind pașii de mai sus linie cu linie, fereastra terminal (cmd) ar trebui să arate ca în imaginea de mai jos.

```

C:\Documents and Settings\Codrina>cd desktop
C:\Documents and Settings\Codrina\Desktop>cd experimentexp12win
C:\Documents and Settings\Codrina\Desktop\ExperimentExp12Win>cd binwin
C:\Documents and Settings\Codrina\Desktop\ExperimentExp12Win\binWin>main
Modular Language written by Dan U Popa, Ro/Haskell Group.
29/aug/2009 - Rodin - Codename:ExperimentExp12
Noutate:Operatorii logici SI &&, SAU !! sunt implementati.
Puteti sa scrieti serii lungi de SAU ori de SI acum.
Am definit si o instructiune alternativa generalizata.
Cross compilat pe Linux. La adapost de virusi !
main: Prelude.<!!>: index too large

C:\Documents and Settings\Codrina\Desktop\ExperimentExp12Win\binWin>

```

Fig.16 (pașii compleți a programului principal)

În rândurile de mai jos vă voi face o demonstrație cum puteți rula un program în Rodin. Scrieti cu ajutorul unui editor de text programul de mai jos, format din doua randuri:

```

{ text „Salut, ai reusit sa rulezi programul.”
}

```

Apoi salvati-l in directorul unde este si „main.exe” sub numele: p2.txt

Tastati in consola> main.exe p2.txt

```

C:\Documents and Settings\Codrina>cd desktop
C:\Documents and Settings\Codrina\Desktop>cd experimentexp12win
C:\Documents and Settings\Codrina\Desktop\ExperimentExp12Win>cd binwin
C:\Documents and Settings\Codrina\Desktop\ExperimentExp12Win\binWin>main.exe
Modular Language written by Dan U Popa, Ro/Haskell Group.
29/aug/2009 - Rodin - Codename:ExperimentExp12
Noutate:Operatorii logici SI &&, SAU !! sunt implementati.
Puteti sa scrieti serii lungi de SAU ori de SI acum.
Am definit si o instructiune alternativa generalizata.
Cross compilat pe Linux. La adapost de virusi !
main.exe: Prelude.<!!>: index too large

C:\Documents and Settings\Codrina\Desktop\ExperimentExp12Win\binWin>main.exe p2.
txt
Modular Language written by Dan U Popa, Ro/Haskell Group.
29/aug/2009 - Rodin - Codename:ExperimentExp12
Noutate:Operatorii logici SI &&, SAU !! sunt implementati.
Puteti sa scrieti serii lungi de SAU ori de SI acum.
Am definit si o instructiune alternativa generalizata.
Cross compilat pe Linux. La adapost de virusi !
Programul:p2.txt
{ text "Salut ai reuit sa rulezi programul"
}
"Salut ai reuit sa rulezi programul"

Programul a rulat !
Modular Language written by Dan U Popa, Ro/Haskell Group.
http://www.haskell.org/haskellwiki/Rodin/Download
29/aug/2009 - Rodin - Codename:ExperimentExp12
C:\Documents and Settings\Codrina\Desktop\ExperimentExp12Win\binWin>

```

Fig.17 (executarea cu succes a primului program în RODIN)

Urmând pașii de mai sus și executând cu main.exe programul scris în pseudocod Rodin obținem

rezultatul executiei. În imagine rulează micul program dat ca exemplu care afișează acest text, stringul "Salut ai reușit să rulezi programul".

**Observație:** Pentru a putea rula programele făcute de voi fișierele \*.txt trebuiesc salvate în dosarul unde se află executabilul programului Rodin sau trebuie sa-i indicati programului Rodin (acel main.exe) numele fisierului .txt cu tot cu directorul in care se afla.

După ce ați rulat programul și ați obținut rezultatul, puteți rula alt program în același mod, numai că în loc de "p2.txt" puneți numele programului vostru "\*.txt", unde "\*" este numele programului pe care voi vreți să îl rulați.

După cum ați observat în **Rodin** programele se salvează ca fișiere text, adică pot fi realizate în orice editor de texte dar de preferat ar fi NotePad sau TotalEdit sau NotePad++.