The Rodin Modular Language vers. 8 aug 2009 – User Manual and Report

Dan POPA, Univ. Lecturer , PhD-C,
Department of Mathematics and Computer Science
University of Bacău, Romania, danvpopa@ub.ro

Abstract: *The scope of this paper is to provide news concerning The Rodin Project (http://www.haskell.org/haskellwiki/Rodin) – a national specific modular didactic language actually used as a helping tool in teaching base of computer science in high school and universities. The problem of producing enough programmers is actual and is a necessary step in order to assure the future development of the IT industry, services and software infrastructure. Rodin is dedicated to the teaching of C-like language's concepts, a wide used set of languages. The Rodin Language is specific designed to cross the language barrier which appears when students without knoledge of English Language are supposed to quickly learn structured programming. Ther Rodin Language was release in aug. 2008. Teachers and students are encoureged and invited to contribute in order to build a corpus of Rodin Programs, based on the model of Free Software Groups. The sources written using Rodin are actually available free of charges from it's website [5]. Rodin is used by The Faculty of Mathematics of Bacau University and also by some high scools from Bacau and Iasi area. The papers contains information concerning several aspects of the project, visible at users level: syntax, examples, differences, notes, how to's .*

This community project dedicated to teachers – The Rodin Language - is presented below.
Keywords: *open source; community; Rodin, C-like languages, structured programming.*

1. Why can programs written using Rodin be classified as an open source initiative

- the free distribution of such programms via website [5], everybody being encouraged to use Rodin and produce good quality teching materials based on it. The Rodin licence is not published yet, but, till this version, Rodin is covered in fact by a sort of BSD licence.
- the source of the Rodin programs are distribuited. Rodin being actually implemented as an interpreter like in [1], it runs sources directly and supports and encourage the study of the sources by reprinting them – on screen – as part of the runing process.
- there are no limitations for derived works, till now, excepting the obligation of mentioning the Rodin webpage and author and also the author of the previous works. Other legal obligations applies too if any.
- every teacher, student, may benefit of the actally published versions of Rodin programs and the released sources.
- By it's design The Rodin Language is made for teaching computer sciens, but we don't forbid any other utilisations of it. If next versions will be good for example for game design or for applications – well – why not ?
- Rodin did not have a specific IDE now and various editors and IDE can be used: Total Edit, Ultra Edit etc. Therefore we did not forbid the integration of the Rodin Interpreter itself and the Rodin Programs with or in any other tools.
- Version of Rodin was build on various platforms: Linux, Wine, Windows (tm). So there is no restriction concerning the O.S. Nobody requested Mac OS versions of Rodin but we are ready to produce a Rodin 4 Mac if needed.
- Rodin is build as a modular interpreter in Haskell [2], [3] – also a free software project. If anybody wants to rebuild Rodin using old technologies like TPLex and TPYacc for example, it is not forbidden, but such person is warned that modularity will be lost, probably gaining

speed instead of it. Because we intend to develop Rodin by adding language modules we are not recommanding to rebuild Rodin on other technologies, but it is not forbidden.

- Rodin syntactic specification can also can be used instead Tiger or While language in The Compiler Construction Course. But, being of the level of the third academic year, it's higly improbable that students did not understand English.
- Rodin module's structure will – probably - be available for those Haskell programmers interested in building language plugins. It's not the case yet. The theories are also published in the Haskell Community [3], also a Open Source Free Software – BSD Style – Community.

2. The Rodin Teacher's Community

- Actually there exist one point where interested teachers may go in order to find Rodin resources: Programs, Teaching Examples/Samples, Syntax specification, News, Advices etc. It is the Rodin Website. Http://www.haskell.org/haskellwiki/Rodin [5] Separated pages are made for: News, Downloads, Questios, Programs etc. The infrastructure is in fact a wiki but the acces is possible only with password, the same provided for [3].
- Teachers are encoureged to built their own sets of Demo Programs in Rodin in order to show the concepts of Structured Programming to their students. We, at Bacau Univ. have also developed some chapters of course for the future teachers in mathematics and basics of Comp. Sci. as part of a Course called : Funadmentals of Programming Languages. They were used during the academic year 2008-2009 and was wellcomed by our students. Such students are and will be the first generation of members of the community.
- As bookwriter I had the idea of placing some of manuals of mine under free licences as Open Documents, in order to be free in using it. They will help us in producing Rodin Language Manuals.
- The starter kit of Rodin consist in a binary interpreter and open/public sources of some programs, showing the main Structured Programming Concepts in Romanian. Students and even kids are able to read such sources, without any knoledge of English. A program like: { citeste x; scrie x } will be easy understand by a native as { read x; write x} withoy problems. The starter kit is provided as an archive containing the Rodin Language and sources.
- Help is provided via Pidgin (an other Open Source Project) using the Yahoo Mail accounts of the users from the community. We even give advices by phone, for the sistem administrators interested in installing Rodin in School's Laboratories.

This set of aspects are supposiong to give you a basic idea concerning the Rodin Teachers's Community.

3. The Rodin Language itself, characteristics of the version 8/08/2009:

Being considered a teaching tool, this version hav some distinctive characteristics:

- it is a small version, intended for beginners, no vectors or records included in this specific version (if neded, use Rodin2009a-e versions, but with caution.)
- during the summer of 2009 the main source of Rodin itself was "sliced" in modules, as part of a research work, also, in order to help the development and the revison. The built of a modular language itself is actually an open research area, but it will be subject of a technical paper.
- a difference: the sintax of sequences was chenged, being now closed to a mixture of Pascal and C. A sequence did not requiered the semicollon after the last statement. The begin and the end are marked with { } as C-like languages does.

- every modul of the parser was triple checked. Teachers can count on a better parser and clear error messages.
- the operators: >=, <=, ==, != are included. Also : ! - the negation
- the "text" command was improved: Special characters like: . , ? ! : = helps user in order to formulate clear messages. Also the @ sign was included in text's specification. The user can program meesages containing e-mail addresses.
- better eror messages, missing keywords are know corectly and completly anounced
- C-like logic: 0 meaning False and other integers meaning true.
- the "let" statement called "fie" remains in place but it's restricted to simple identifiers – on the left side.
- the name of the running program is also sent to the console output.
- if really needed, the sequences of statements may be separated by « , » too, not only by semicollon.
- commands and expressions are know separated syntactic categories
- the "for" statement called "pentru" has a new syntax:

```
pentru (<com> ; <exp>;  <com>)
     <com>
```

Removed characteristics: Don't count on concepts like:

- vectors, indexed variables
- the "let" statement called "fie" where the left side is an indexed variable
- records – was not implemented at all, in any version
- files – also not implemented
- anonymous 1-parameter functions expresed as abstractions
- the apply invisible/unwritten operator
- the vide sequence {}

4. The Rodin Language itself, syntax of the version 8/08/2009, codename:ExperimentExp8

4.1 The **I/O operations** are,yet, console based. There exists a sort of "read" called "citeste", a sort of "write" called "scrie", and also a sort of "writeStr" called "text". Examples:

citeste x
scrie y
text "dati valoarea lui x:"

The strings mai contain letters , digits and some extra characters, very helpfull in order to make simple sentences: ! ? , . = @ _ - or to speak about e-mail adresses.

4.2 **Assignments**: Values are assigned to variables using a "let" statement as in Basic. It's syntax is : let <var> = <exp> where the expression may contains any kind of operators: +, -, * , /, > , < , >=, <=, !=, ==, ! .

```
fie x=1;
fie y=x+1;
fie z=(x+1)*(y+2);
fie logic=(z<=10);
fie negat=! (z<0);
```

```
Nota: && si OR nu sunt implementati in aceasta versiune.
```

4.3. The **"begin... end"** block statement is replaced by "{ ......   }", where single statements can pe separated using ";" and also "," (not recommanded but also possible).

```
{citeste x;
 scrie x }

Notati:Nu este permis spatiul de dupa "cand".
```

```
{text "dati valoarea lui x:";
 citeste x;
 scrie x }

Notati:Nu este permis spatiul de dupa "cand".
```

Some programs using the translated version of the "begin ... end" sequence, inspired by C-like languages..

4.4. The **"if" "then" "else"** becomes "daca" "atunci" "altfel". A simplified version: The "if" "then" becomed "daca" "atunci" and it also usable.

```
{ daca (1==1) atunci scrie 10 altfel scrie 0 }

-- daca1.txt
-- Comparatii: egalitatea scrisa cu 2 de egal
```

```
{ citeste x;
  citeste y;
  daca (x==y) atunci scrie 10 altfel scrie 0 }

-- daca2.txt
Se pot compara si variabilele, si expresiile...
Orice expresie intreaga poate fi conditie.
```

```
{ citeste x;
  citeste y;
  daca (y!=0) atunci scrie x/y altfel scrie 0 }

-- daca3.txt
Comparatia "diferit" scrisa in stil C.
Impartirea intreaga /.
Se pot compara si variabilele, si expresiile...
Orice expresie intreaga poate fi conditie.
```

```
{text "Dati urmatorul y ";
citeste y;
text "Dati urmatorul xm ";
citeste xm;
executa {
   {daca (y>xm)
    atunci fie xm=y    };
atat cat (y!=0);
}
```

```
{text "Start program: dati x, ENTER, y si ENTER";
 citeste x;
 citeste y;
 daca (x>y) atunci text "x mai mare ca y"
 altfel text "x mai mic sau egal cu y";
 text " apasa 0 si Enter";
 citeste z
 }
```

Modular Language written by Dan V Popa, Ro/Haskell Group.
8/aug/2009 - Rodin - Codename:ExperimentExp8
limitare :{ <com> ; <com> ... <com> } fara ; final.

Some Programs using the alternative (i.e. Conditional) statement.

4.5. The **"while"** keyword is replaced by "cat timp". Spaces are allowed between the two keywords. The space between the second keyword and the block of statements, theoretically accepted is not allowed in the actual implementation.

```
{citeste x;
 cat timp(x>0)
   { fie x = x /2;
     scrie x }
}
```

Un numar este impartit repetat la 2.
Rodin Modular / 8.08.2009/ ExperimentExp8
Atentie, aceasta versiune de while nu mai are "executa".
Notati:Nu este permis spatiul de dupa "timp".

```
{ fie x=100;
  cat timp(x>10)
    fie x=x-1;
  scrie x;
  text "Salut!"
}
```

Nu puneti spatiu dupa "timp".
Nu-l va accepta.
Revizuiti sursele vechi.

```
{ text "Calculul lui n! pentru n= ...";
  citeste n;
  fie x=1;
  fie nr=1;
  cat timp(nr<n)
    { fie nr=nr+1;
      fie x=x*nr
    };
  scrie x
}

Modular Language written by Dan V Popa, Ro/Haskell Group.
8/aug/2009  -      Rodin       - Codename:ExperimentExp8
limitare :{ <com> ; <com> ... <com> }  fara ; final.
Programul:RodinV08-Factorial-Ro.txt
```

```
{ fie y=2;
  fie x=100;
  cat timp(x>10) {
    fie x=x-1;
    scrie x
  };
  scrie y;
  text "Salut!"
}

Numaratoarea descendenta:
Bucla cu test initial cu
mai multe instructiuni  in bucla.
```

```
{ citeste n;
  fie f1=0;
  fie f2=1;
  scrie f1;
  scrie f2;
  cat timp(f2<n)
    { fie f1p=f2;
      fie f2p=f1+f2;
      fie f1=f1p;
      fie f2=f2p;
      scrie f1
    }
}

-- 7 aug 2009. Fibo.
-- Refacut cu ocazia Exp 07
-- fara spatiu dupa timp(
-- fara ; dupa ultima instructiune
```

Some Programs using the ro-version of the "while" loop.

4.6. The **"do... while ..."** statement is replaced by "executa.... atat cat ". Spaces are allowed between the two keywords. The space between the second keyword and the expression, theoretically accepted are not allowed in the actual implementation.

```
{ text "  Maximul elementelor unui sir de numere ";
  text "pozitive distincte terminat cu numarul zero. ";
  fie xmax = 0;
  text "dati y ";
  citeste y;
  executa {
   {daca (y>xmax)
    atunci fie xmax=y
   };
   text "dati urmatorul y ";
   citeste y }
  atat cat (y!=0);
  text "maximul este ";
  scrie xmax
}

--Rev 9 aug 2009 pt ExperimentExp8
--Instructiunea
                              executa ... atat cat ...
Este echivalentul lui do... ..........while ... din C.
Primul loc: o instructiune (compusa eventual)
Al doilea: conditia

-- Instructiunea daca ... atunci...
fara alternativa:altfel
```

A Program using the translated version of the "do... while ...." loop, which is specific for the C-like languages.

4.7. The **"for"** keyword is replaced by "pentru". Dual and multiple counters loops are allowed.

```
{pentru (fie x=1;  x<10;  fie x=x+1)
   scrie x
}

--Rodin, 8 aug 2009, Exp8
```

```
{pentru (
        {fie x=1,fie y=2};
        x*x<100;
        {fie x=x+1,fie y=y*2}
        )
          {text "x=";
           scrie x;
           text "y=";
           scrie y;
           text " "}
}

  Modular Language written by Dan V Popa, Ro/Haskell Group.
8/aug/2009  -      Rodin       - Codename:ExperimentExp8

  Programul:bucladubla.txt.
La instructiunea for este nevoie de acolade la comenzile
c1, c3, c4 din
  for ( c1 ; e2 ; c3 ) c4
Se pot scrie si acele ciudate bucle cu doua contoare.
```

Some programs using the translated version of the "for" loop.

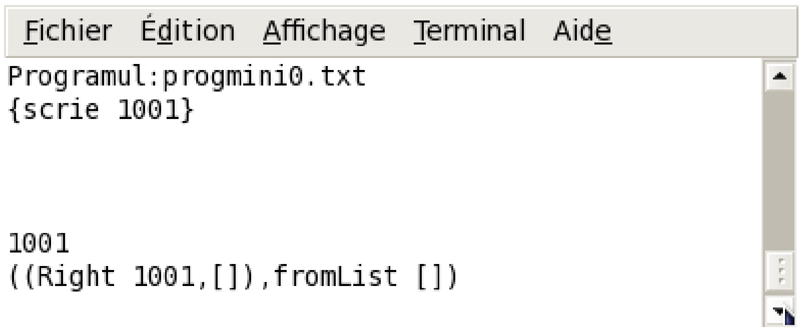
4.8. The **"repeat... until ....."** statement is replaced by "repeta ...pina cand.....".

```
{citeste x;
 repeta
   { fie x = x /2;
     scrie x }
 pina cand(x==0)
}

Rodin Modular / 8.08.2009/ ExperimentExp8
Notati:Nu este permis spatiul de dupa "cand".
```

```
{ text "Calculul divizorului comun";
  text "dati numarul a ";
  citeste a;
  text "dati numarul b ";
  citeste b;
  fie undeimp=a;
  fie unimp=b;
  repeta
    { fie unrest=undeimp%unimp;
      fie undeimp=unimp;
      fie unimp=unrest
    }
  pina cand (unimp==0);
  text "Iata divizorul comun:";
  scrie undeimp
};


Program 4:cmmdc
==================
Modular Language written by Dan V Popa, Ro/Haskell Group.
8/aug/2009  -      Rodin       - Codename:ExperimentExp8
```

Some programs using the translated version of the "for" loop.

5: Running programs.



```
Programul:progmini0.txt
{scrie 1001}



1001
((Right 1001,[]),fromList [])
```

The Rodin programs are stored in common texts files and can be edited with any editor supporting the txt file format. Simply run the Main Rodin Binary from a console or from the menu of the editor, where can be easily added (Ultra Edit, Total Edit, X Emacs etc...).The name of the program is given as a single parameter. The same procedure is used on various operating systems: Window, Linux etc.

6. Arithmetics.

Because Rodin inherits it's long arithmetics from it's development language Haskell [2] and the integers implementation was made via the data declaration [2], Cap 3 , pg 57-66 using Integers (the type of long integers available in Haskell), Rodin can also manipulate long arithmetics.

Here is a program used to computed 100! (and the previous values , too). The trace list of values is also printed:

```
[dan@localhost ExperimentExp8]$ ./Main  factorial.txt
Modular Language written by Dan V Popa, Ro/Haskell Group.
8/aug/2009  -       Rodin        - Codename:ExperimentExp8
limitare :{ <com> ; <com> ... <com> }  fara ; final.
Programul:factorial.txt
{ citeste n;
  fie x=1;
  fie nr=1;
  cat timp(nr<n)
    { fie nr=nr+1;
      fie x=x*nr
    };
  scrie x
}


100

933262154439441526816992388562667004907159682643816214685929638952
175999932299156089414639761565182862536979208272237582511852109168
64000000000000000000000000
((Right
933262154439441526816992388562667004907159682643816214685929638952
175999932299156089414639761565182862536979208272237582511852109168
64000000000000000000000000,[" n=100"," x=1"," nr=1"," nr=2","
x=2"," nr=3"," x=6"," nr=4"," x=24"," nr=5"," x=120"," nr=6","
x=720"," nr=7"," x=5040"," nr=8"," x=40320"," nr=9"," x=362880","
nr=10"," x=3628800"," nr=11"," x=39916800"," nr=12","
x=479001600"," nr=13"," x=6227020800"," nr=14"," x=87178291200","
nr=15"," x=1307674368000"," nr=16"," x=20922789888000"," nr=17","
x=355687428096000"," nr=18"," x=6402373705728000"," nr=19","
x=121645100408832000"," nr=20"," x=2432902008176640000"," nr=21","
x=51090942171709440000"," nr=22"," x=1124000727777607680000","
nr=23"," x=25852016738884976640000"," nr=24","
x=620448401733239439360000"," nr=25","
x=15511210043330985984000000"," nr=26","
x=403291461126605635584000000"," nr=27","
x=10888869450418352160768000000"," nr=28","
x=304888344611713860501504000000"," nr=29","
```

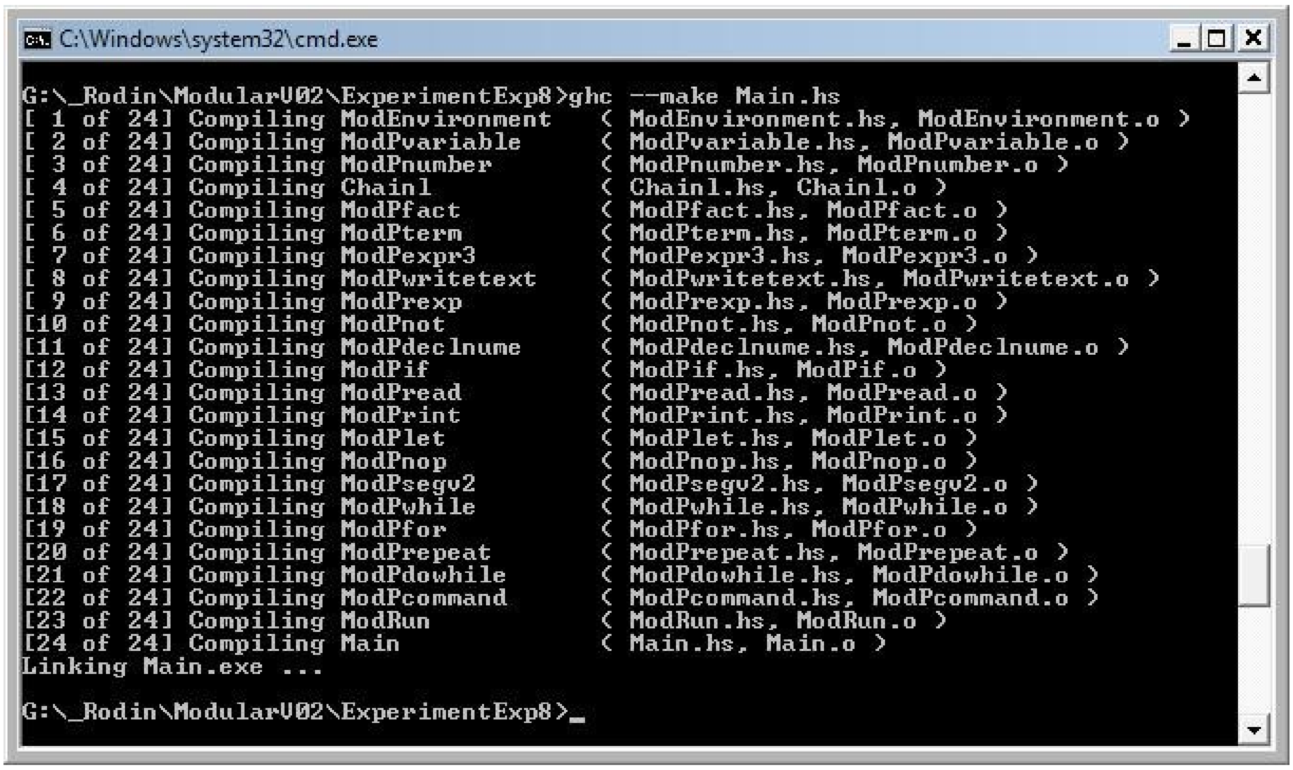
```
x=8841761993739701954543616000000"," nr=30","
x=265252859812191058636308480000000"," nr=31","
x=8222838654177922817725562880000000"," nr=32","
x=263130836933693530167218012160000000"," nr=33","
x=8683317618811886495518194401280000000"," nr=34","
x=295232799039604140847618609643520000000"," nr=35","
x=10333147966386144929666651337523200000000"," nr=36","
x=371993326789901217467999448150835200000000"," nr=37","
x=13763753091226345046315979581580902400000000"," nr=38","
x=523022617466601111760007224100074291200000000"," nr=39","
x=20397882081197443358640281739902897356800000000"," nr=40","
x=815915283247897734345611269596115894272000000000"," nr=41","
x=33452526613163807108170062053440751665152000000000"," nr=42","
x=1405006117752879898543142606244511569936384000000000"," nr=43","
x=60415263063373835637355132068513997507264512000000000","
nr=44","
x=2658271574788448768043625811014615890319638528000000000","
nr=45","
x=119622220865480194561963161495657715064383733760000000000","
nr=46","
x=5502622159812088949850305428800254892961651752960000000000","
nr=47","
x=258623241511168180642964355153611979969197632389120000000000","
nr=48","
x=12413915592536072670862289047373375038521486354677760000000000",
" nr=49","
x=608281864034267560872252163321295376887552831379210240000000000"
," nr=50","
x=3041409320171337804361260816606476884437764156896051200000000000
0"," nr=51","
x=1551118753287382280224243016469303211063259720016986112000000000
000"," nr=52","
x=8065817517094387857166063685640376697528950544088327782400000000
0000"," nr=53","
x=4274883284060025564298013753389399649690343788366813724672000000
000000"," nr=54","
x=2308436973392413804720927426830275810832785645718079411322880000
00000000"," nr=55","
x=1269640335365827592596510084756651695958032105144943676227584000
0000000000"," nr=56","
x=7109985878048634518540456474637249497364979788811684586874470400
00000000000"," nr=57","
x=4052691950487721675568060190543232213498038479622660214518448128
0000000000000"," nr=58","
x=2350561331282878571829474910515074683828862318181142924420699914
240000000000000"," nr=59","
x=1386831185456898357379390197203894063459028767726874325408212949
40160000000000000"," nr=60","
x=8320987112741390144276341183223364380754172606361245952449277696
409600000000000000"," nr=61","
x=5075802138772247988008568121766252272260045289880360030994059394
80985600000000000000"," nr=62","
x=3146997326038793752565312235495076408801228079725823219216316824
```

```
78211072000000000000000"," nr=63","
x=1982608315404440064116146708361898137544773690227268628106279599
61272975360000000000000000"," nr=64","
x=1268869321858841641034333893351614808028655161745451921988018943
752147042304000000000000000"," nr=65","
x=8247650592082470666723170306785496252186258551345437492922123134
3889557749760000000000000000"," nr=66","
x=5443449390774430640037292402478427526442930643887988745328601268
69671081148416000000000000000"," nr=67","
x=3647111091818868528824985909660546442716763531404952459370162850
026796243694387200000000000000000"," nr=68","
x=2480035542436830599600990418569171581047399201355367672371710738
0182214457121832960000000000000000"," nr=69","
x=1711224524281413113724683388812728390922705448935203693936480409
232572797541406474240000000000000000"," nr=70","
x=1197857166996989179607278372168909873645893814254642585755536286
4628009582789845319680000000000000000"," nr=71","
x=8504785885678623175211676442399260102885846081207962358864307633
88588680378079017697280000000000000000"," nr=72","
x=6123445837688608686152407038527467274077809178469732898382301496
397838498722168927420416000000000000000000"," nr=73","
x=4470115461512684340891257138125051110076800700282905015819080092
3704221040671833170169036800000000000000000"," nr=74","
x=3307885441519386412259530282212537821456832518209349711706119268
35411235700971565459250872320000000000000000000"," nr=75","
x=2480914081139539809194647711659403366092624388657012283779589451
265584267757286740944381542400000000000000000000"," nr=76","
x=1885494701666050254987932260861146558230394535379329335672487982
961844043495537923117729972224000000000000000000000"," nr=77","
x=1451830920282858696340707840863082849837403792242083588467815746
88061991349156420080065207861248000000000000000000000"," nr=78","
x=1132428117820629783145752115873204622873174957948825199004896282
566883532523420076624508621317734400000000000000000000"," nr=79","
x=8946182130782975286851441715398316520698082167795719072138680632
278379906935018605333618108410101760000000000000000000000"," nr=80","
x=7156945704626380229481153372318653216558465734236575257710944505
82270392554801488426689448672808140800000000000000000000","
nr=81","
x=5797126020747367985879734231578109105412357244731625958745865049
716390179693892056256184534249745940480000000000000000000000","
nr=82","
x=4753643337012841748421382069894049466438132940679933286171609340
76743994734899148613007131808479167119360000000000000000000000","
nr=83","
x=3945523969720658651189747118012061057143650340764344627522435752
836975156299662933487959194010377087090688000000000000000000000","
nr=84","
x=3314240134565353266999387579130131288000666286242049487118846032
38305913129171686412988572296871675315617792000000000000000000000","
nr=85","
x=2817104114380550276949479442260611594800566343305742064051019127
525600261597959334510402864523409240182751232000000000000000000000"
," nr=86","
```

```
x=242270953836727323817655232034412597152848705524293817508387644967201622497424502767894646349013194655716605952000000000000000000
0"," nr=87","
x=210775729837952771721360051869938959522978373806135621232297251121465411572759317408068342323641479350473447178240000000000000000
000"," nr=88","
x=185482642257398439114796845645546284380220968949399346684421580986889562184028199319100141244804501828416633516851200000000000000
00000"," nr=89","
x=165079551609084610812169192624536193098396662364965418549135207078331710343785097393999125707876006627290803829997568000000000000
0000000"," nr=90","
x=148571596448176149730952273362082573788556996128468876694221686370498539309406587654599213137088405964561723446997811200000000000
000000000"," nr=91","
x=135200152767840296255166568759495142147586866476906677791741734597153670771559994765685283954750449427751168336768008192000000000
00000000000"," nr=92","
x=124384140546413072554753243258735530775779917158754143568402395829381377109835195184430461238370413473531074869826567536640000000
0000000000000"," nr=93","
x=115677250708164157475920516230624043621475322957641353518614228121324680712146731521520328951684484530383899628938707809075200000
000000000000000"," nr=94","
x=108736615665674308027365285256786601004186803580182872307497374434045199869417927630229109214583415458560865651202385340530688000
00000000000000000"," nr=95","
x=103299784882390592625997020993947270953977463401173728692122505712342939875947031248717653753854244685632822368642266073504153600
0000000000000000000"," nr=96","
x=991677934870949689209571401541893801158183648651267795444376054838492222809091499987689476037000748982075094738965754305639874560
0000000000000000000"," nr=97","
x=961927596824821198533284259495636987123438139191729761581044773193337456124818754988058791755890726512612841896796781676470678323
2000000000000000000000"," nr=98","
x=942689044888324774562618574305724247380969376407895166349423877729470707002322379888297615920772911982360585058860846042941264756
7360000000000000000000000"," nr=99","
x=933262154439441526816992388562667004907159682643816214685929638952175999932299156089414639761565182862536979208272237582511852109
1686400000000000000000000000"," nr=100","
x=933262154439441526816992388562667004907159682643816214685929638952175999932299156089414639761565182862536979208272237582511852109
168640000000000000000000000000"]),fromList [("n",100),("nr",100),
("x",933262154439441526816992388562667004907159682643816214685929
6389521759999322991560894146397615651828625369792082722375825118521
09168640000000000000000000000000)])

Sfarsit !
Modular Language written by Dan V Popa, Ro/Haskell Group.
8/aug/2009  -            - Codename:ExperimentExp8
[dan@localhost ExperimentExp8]$
```

7. Conclusions

This article is dedicated to The Rodin Community, a community of teachers dedicated to make C-like languages affordable by Romanian Students. The article focuses on the latest stage of development of the Rodin Langage, which had been revised during this summer of 2009. The Rodin version of the moment (after one year from it's first release in 2008) is a bit different – being modulalrly sliced and verified module by module – and then rebuild on different platforms.



C:\Windows\system32\cmd.exe

```
G:\_Rodin\ModularV02\ExperimentExp8>ghc --make Main.hs
[ 1 of 24] Compiling ModEnvironment   ( ModEnvironment.hs, ModEnvironment.o )
[ 2 of 24] Compiling ModPvariable     ( ModPvariable.hs, ModPvariable.o )
[ 3 of 24] Compiling ModPnumber       ( ModPnumber.hs, ModPnumber.o )
[ 4 of 24] Compiling Chainl           ( Chainl.hs, Chainl.o )
[ 5 of 24] Compiling ModPfact         ( ModPfact.hs, ModPfact.o )
[ 6 of 24] Compiling ModPterm         ( ModPterm.hs, ModPterm.o )
[ 7 of 24] Compiling ModPexpr3        ( ModPexpr3.hs, ModPexpr3.o )
[ 8 of 24] Compiling ModPwritetext    ( ModPwritetext.hs, ModPwritetext.o )
[ 9 of 24] Compiling ModPrexp         ( ModPrexp.hs, ModPrexp.o )
[10 of 24] Compiling ModPnot          ( ModPnot.hs, ModPnot.o )
[11 of 24] Compiling ModPdeclnume     ( ModPdeclnume.hs, ModPdeclnume.o )
[12 of 24] Compiling ModPif           ( ModPif.hs, ModPif.o )
[13 of 24] Compiling ModPread         ( ModPread.hs, ModPread.o )
[14 of 24] Compiling ModPrint         ( ModPrint.hs, ModPrint.o )
[15 of 24] Compiling ModPlet          ( ModPlet.hs, ModPlet.o )
[16 of 24] Compiling ModPnop          ( ModPnop.hs, ModPnop.o )
[17 of 24] Compiling ModPsegv2        ( ModPsegv2.hs, ModPsegv2.o )
[18 of 24] Compiling ModPwhile        ( ModPwhile.hs, ModPwhile.o )
[19 of 24] Compiling ModPfor          ( ModPfor.hs, ModPfor.o )
[20 of 24] Compiling ModPrepeat       ( ModPrepeat.hs, ModPrepeat.o )
[21 of 24] Compiling ModPdowhile      ( ModPdowhile.hs, ModPdowhile.o )
[22 of 24] Compiling ModPcommand      ( ModPcommand.hs, ModPcommand.o )
[23 of 24] Compiling ModRun           ( ModRun.hs, ModRun.o )
[24 of 24] Compiling Main             ( Main.hs, Main.o )
Linking Main.exe ...

G:\_Rodin\ModularV02\ExperimentExp8>_
```

Building the modular interpreter of Rodin from the same Haskell package of sources, on a Windows Vista Home Basic Platform.

The main theoretic aspects of Rodin as those presented on Anglo Haskell 2008 web page from [3] by myself will be subject of an other paper or on other book like [1].

8. References

[1] – Dan Popa, Practica Interpretarii Monadice, MatrixRom, Bucuresti, 2008,
ISBN 978-973-755-417-8
[2] - Dan Popa, Introducere in Haskell 98 prin exemple , Edusoft , Bacau, 2007,
ISBN 978-973-8934-48-1
[3] - The Haskell Org Community – www.haskell.org
[4] - The Ro/Haskell Community – www.haskell.org/haskellwiki/Ro/Haskell
[5] - The Rodin Community – www.haskell.org/haskellwiki/Rodin
[6] - Pidgin – http://www.pidgin.im

Authors' details: Dan Popa is University Lecture at Mathematics and ComputerScience Department from Faculty of Sciences of the University of Bacau. He is a PhD candidate from December 2001 at Informatics Department from "Al.I.Cuza" University, Iasi. His main research areas related to this paper are:

modular monadic parsing and semantics. Founder of the Ro/Haskell Community [4] and originator of The Rodin Project [5]. He is a HCAR corespondent for Romania, too.