

# اصول برنامه Glomosisim

اقتباس از Michael AuYeung و چند منبع دیگر  
ترجمه: مهدی عبداللهی

## مقدمه

- GloMoSim مجموعه ای از توابع کتابخانه ای جهت شبیه سازی سیستم های اطلاعاتی سیار می باشد که به طور خاص برای شبیه سازی شبکه های بی سیم و تک منظوره به کار می رود. این نرم افزار از طریق آدرس وب دانشگاه یوسی ال ای آمریکا قابل دریافت است.

- <http://pcl.cs.ucla.edu/projects/glomosim/>

- کاربری این نرم افزار نسبت به NS - که یک نرم افزار شبیه سازی دیگر است - آسان تر است.

# فهرست مطالب

- نحوه نصب روی سیستم عامل های بر مبنای یونیکس
  - مثال در مورد RPM-based Linux (RPM) یک ابزار خط فرمانی قوی برای نصب، حذف و ارتقای نرم افزار های کامپیوتری است)
- نصب روی سیستم عامل ویندوز
- اجرای یک شبیه سازی ساده
- مشاهده نتایج در GloMoSim VT

# نصب در سیستم عامل بر مبنای یونیکس

• ما مراحل نصب را در سیستم لینوکس بر مبنای RPM نشان می دهیم:

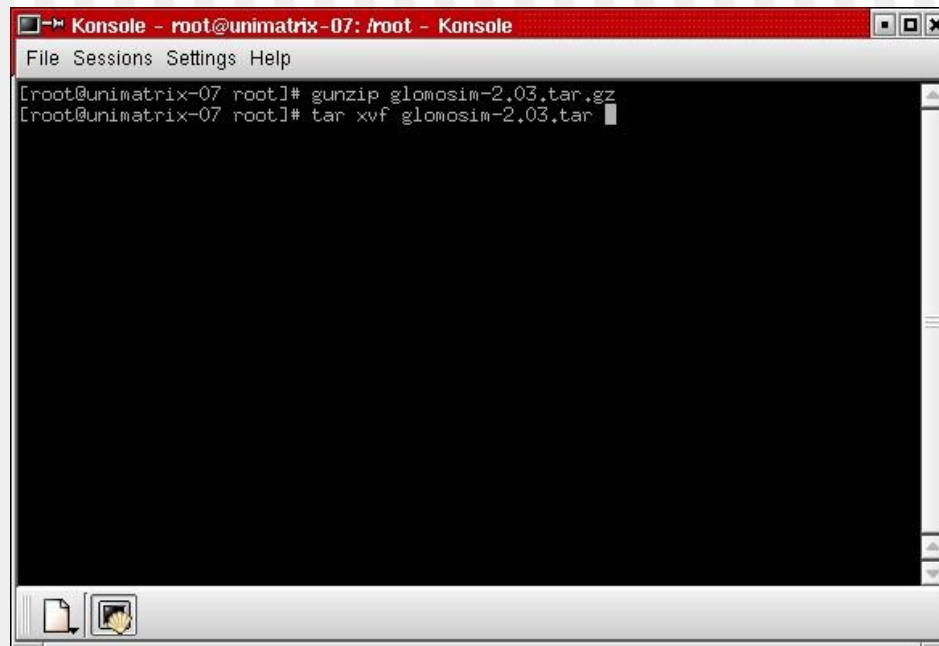
- Red Hat
- Mandrake
- ...

• با فرمان زیر بسته را از حالت فشرده خارج می کنیم:

• **gunzip glomosim-2.03.tar.gz**

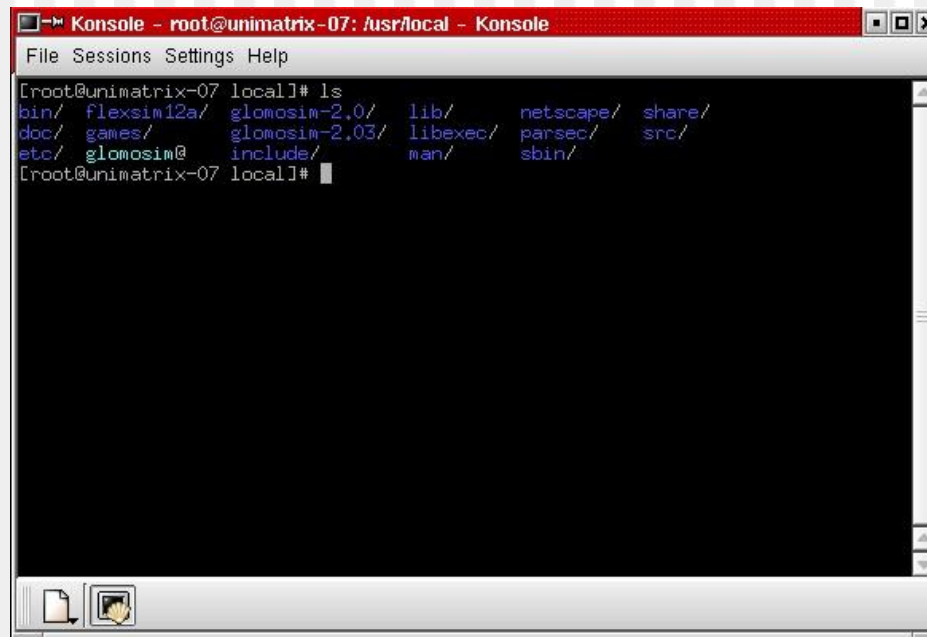
• با دستور زیر بسته را باز می کنیم:

• **tar xvf glomosim-2.03.tar**



```
Konsole - root@unimatrix-07: /root - Konsole
File Sessions Settings Help
[root@unimatrix-07 root]# gunzip glomosim-2.03.tar.gz
[root@unimatrix-07 root]# tar xvf glomosim-2.03.tar
```

- glomosim و parsec را در پوشه /usr/local نصب می کنیم.

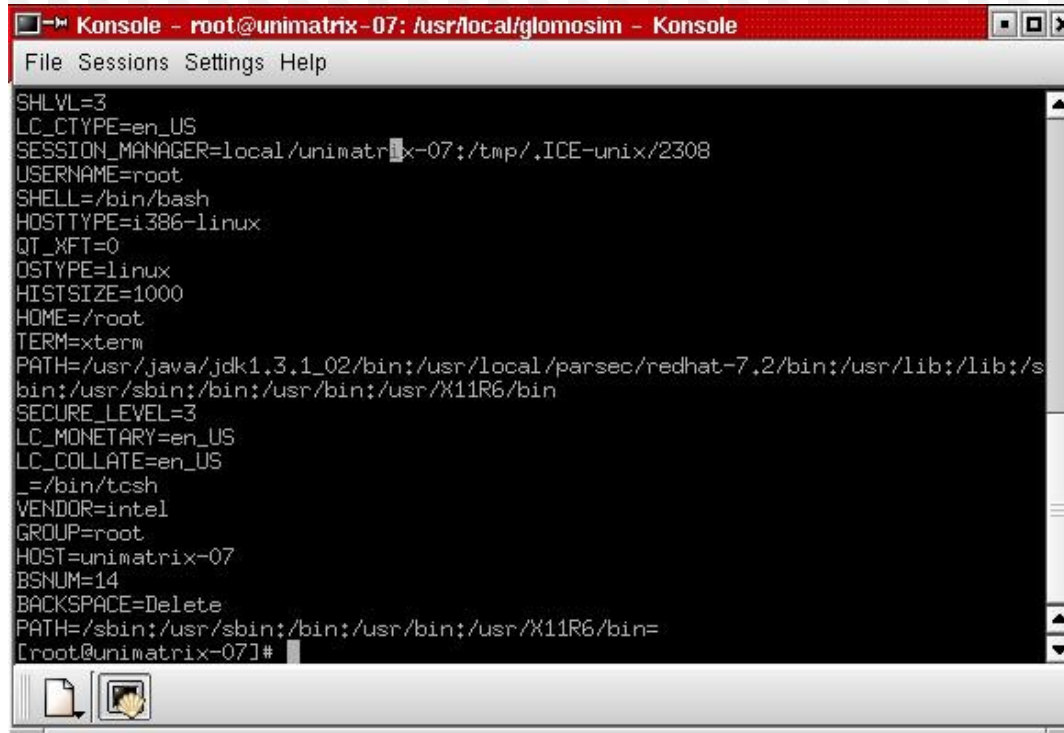


```
Konsole - root@unimatrix-07: /usr/local - Konsole
File Sessions Settings Help
[root@unimatrix-07 local]# ls
bin/ flexsim12a/ glomosim-2.0/ lib/ netscape/ share/
doc/ games/ glomosim-2.03/ libexec/ parsec/ src/
etc/ glomosim@ include/ man/ sbin/
[root@unimatrix-07 local]#
```

- در نظر داشته باشید که پوشه های داخل parsec در واقع فهرست سیستم عامل های پشتیبانی شده توسط برنامه GloMoSim را نشان می دهند. GloMoSim دارای یک سورس کد واحد است که کامپایلر آن روی سیستم عامل های AIX، IRIS، Solaris، Windows NT، Linux و FreeBSD پشتیبانی می شود.

• با فرض استفاده از `tsh` یا `cs` متغیر محیطی `path` را مقدار دهی کنید:

- `PCC_DIRECTORY`
- `PATH`



```
Konsole - root@unimatrix-07: /usr/local/glomosim - Konsole
File Sessions Settings Help
SHLVL=3
LC_CTYPE=en_US
SESSION_MANAGER=local/unimatrix-07:/tmp/.ICE-unix/2308
USERNAME=root
SHELL=/bin/bash
HOSTTYPE=i386-linux
QT_XFT=0
OSTYPE=linux
HISTSIZE=1000
HOME=/root
TERM=xterm
PATH=/usr/java/jdk1.3.1_02/bin:/usr/local/parsec/redhat-7.2/bin:/usr/lib:/lib:/s
bin:/usr/sbin:/bin:/usr/bin:/usr/X11R6/bin
SECURE_LEVEL=3
LC_MONETARY=en_US
LC_COLLATE=en_US
_=/bin/tsh
VENDOR=intel
GROUP=root
HOST=unimatrix-07
BSNUM=14
BACKSPACE=Delete
PATH=/sbin:/usr/sbin:/bin:/usr/bin:/usr/X11R6/bin=
[root@unimatrix-07]#
```

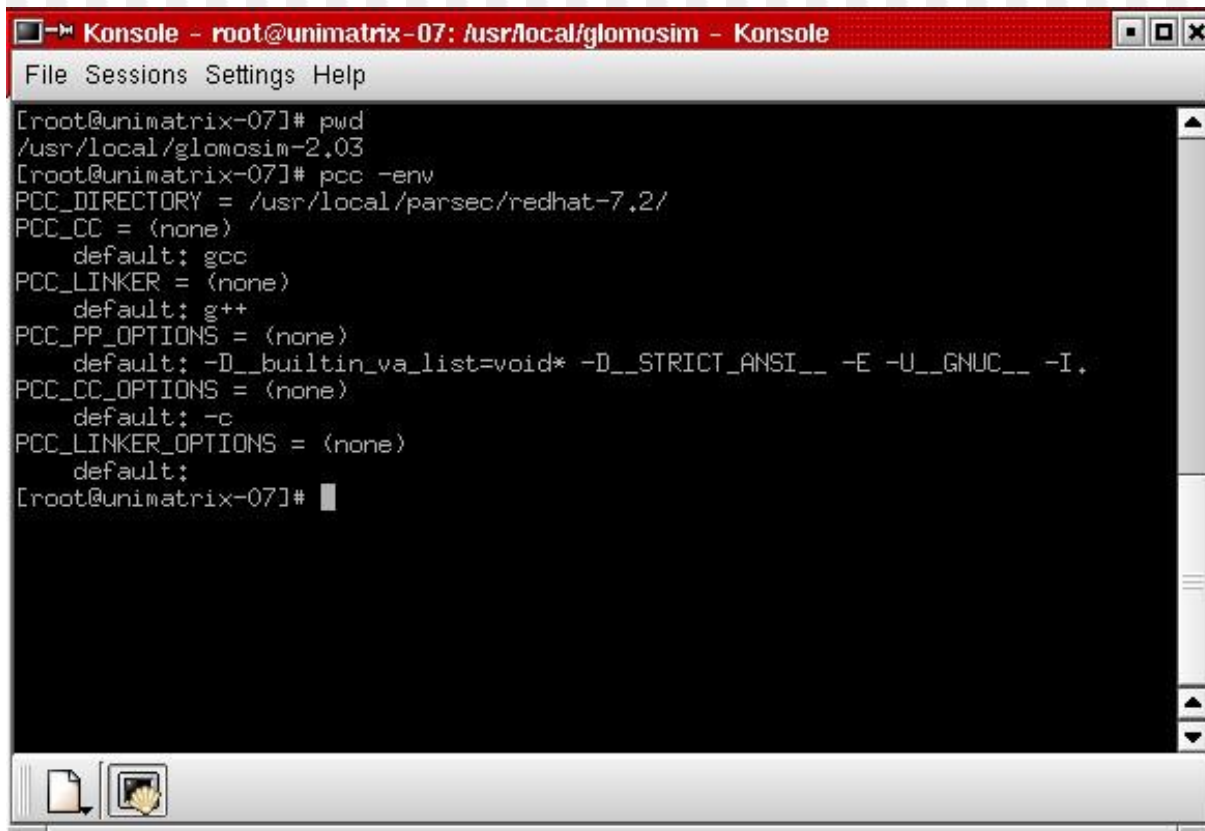


- مقدار `PCC_DIRECTORY` باید برابر `/usr/local/parsec/redhat-7.2` باشد که شما `parsec` را در آن قرار می دهید. اگر شما از لینوکس و `gcc` نگارش ۲.۹۵ یا بالاتر استفاده می کنید استفاده از پوشه `redhat-7.2` الزامی است.

- لازم است که متغیر `PATH` مسیر کامل `pcc` یعنی `/usr/local/parsec/redhat-7.2/bin` را در بر داشته باشد.

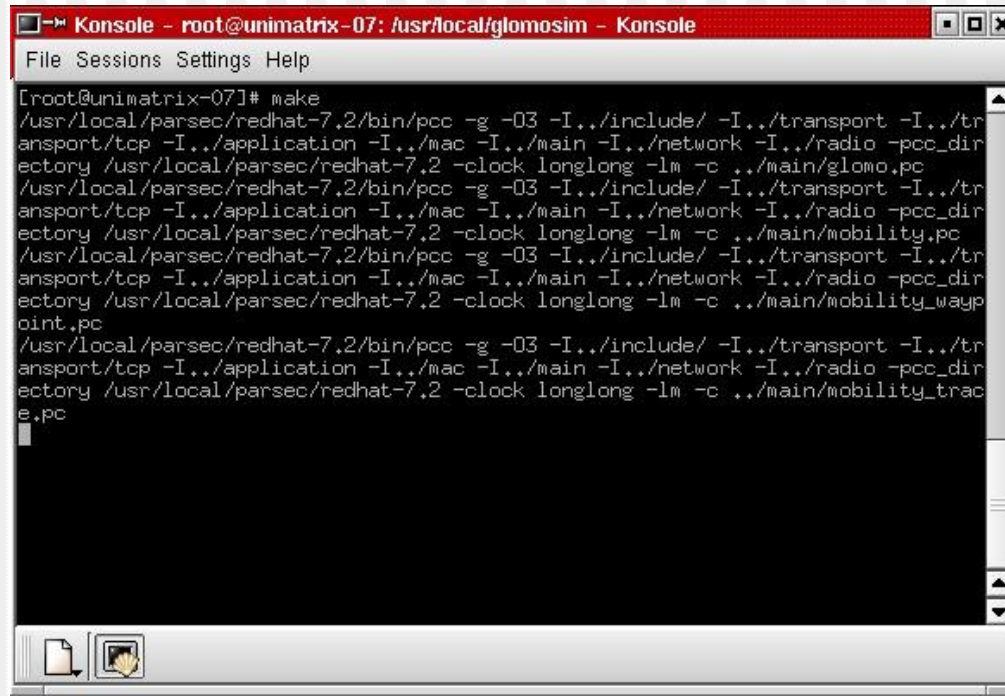
- تنها متغیر محیطی مهم `PCC_DIRECTORY` است صرفنظر از این که مقادیر پیش فرض قابل قبول هستند.

- بررسی کنید که pcc از داخل هر پوشه ای اجرا شود.
- مقدار متغیر محیطی pcc را با فرمان `pcc -env` بررسی نمایید.



```
Konsole - root@unimatrix-07: /usr/local/glomosim - Konsole
File Sessions Settings Help
[root@unimatrix-07]# pwd
/usr/local/glomosim-2.03
[root@unimatrix-07]# pcc -env
PCC_DIRECTORY = /usr/local/parsec/redhat-7.2/
PCC_CC = (none)
    default: gcc
PCC_LINKER = (none)
    default: g++
PCC_PP_OPTIONS = (none)
    default: -D__builtin_va_list=void* -D__STRICT_ANSI__ -E -U__GNUC__ -I.
PCC_CC_OPTIONS = (none)
    default: -c
PCC_LINKER_OPTIONS = (none)
    default:
[root@unimatrix-07]#
```

- از داخل پوشه `glomosim\main` دستور `make` را اجرا کنید.
- پس از اتمام اجرای دستور `make` برنامه `glomosim` از داخل پوشه `bin` قابل اجرا است.
- فایل های مربوط به رابط گرافیکی جاوا را نیز از داخل `java_gui` کامپایل کنید.



```
Konsole - root@unimatrix-07: /usr/local/glomosim - Konsole
File Sessions Settings Help
[root@unimatrix-07]# make
/usr/local/parsec/redhat-7.2/bin/pcc -g -O3 -I../include/ -I../transport -I../tr
ansport/tcp -I../application -I../mac -I../main -I../network -I../radio -pcc_dir
ectory /usr/local/parsec/redhat-7.2 -clock longlong -lm -c ../main/glomo.pc
/usr/local/parsec/redhat-7.2/bin/pcc -g -O3 -I../include/ -I../transport -I../tr
ansport/tcp -I../application -I../mac -I../main -I../network -I../radio -pcc_dir
ectory /usr/local/parsec/redhat-7.2 -clock longlong -lm -c ../main/mobility.pc
/usr/local/parsec/redhat-7.2/bin/pcc -g -O3 -I../include/ -I../transport -I../tr
ansport/tcp -I../application -I../mac -I../main -I../network -I../radio -pcc_dir
ectory /usr/local/parsec/redhat-7.2 -clock longlong -lm -c ../main/mobility_wapp
oint.pc
/usr/local/parsec/redhat-7.2/bin/pcc -g -O3 -I../include/ -I../transport -I../tr
ansport/tcp -I../application -I../mac -I../main -I../network -I../radio -pcc_dir
ectory /usr/local/parsec/redhat-7.2 -clock longlong -lm -c ../main/mobility_trac
e.pc
```

# تصویر خروجی برنامه برای یک پروژه شبیه سازی

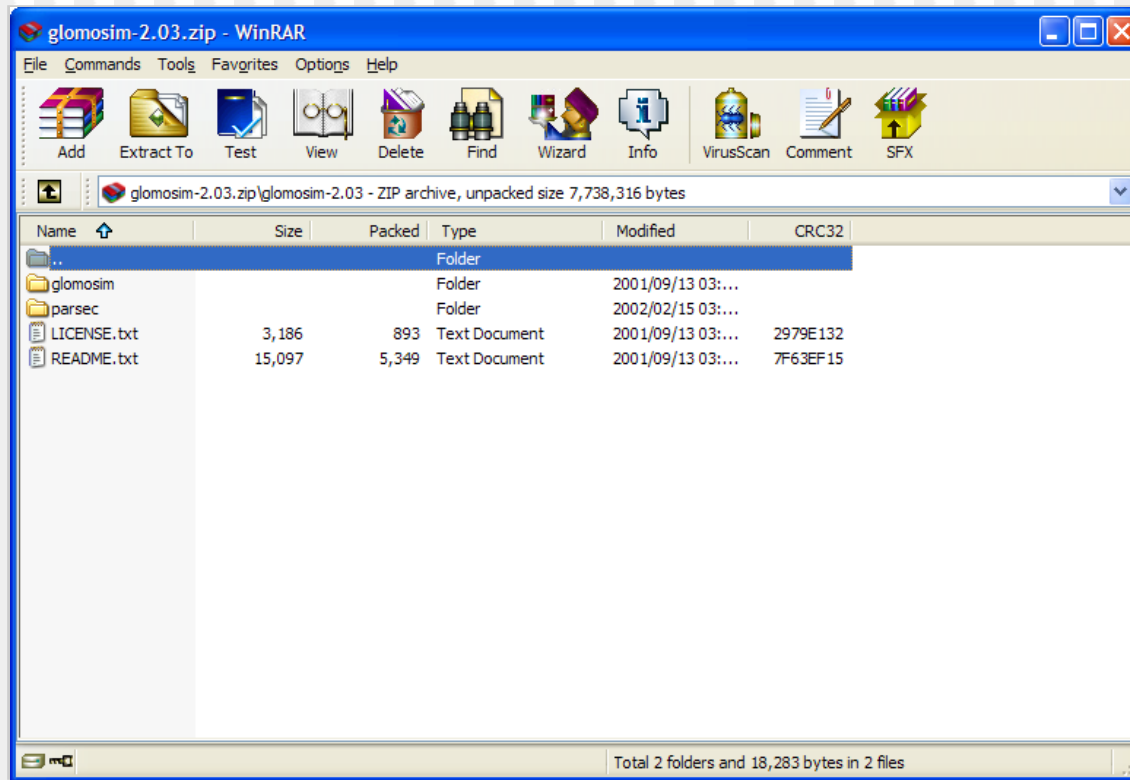
```
Konsole - root@unimatrix-07: /usr/local/gnomosim - Konsole
File Sessions Settings Help
Current Sim Time[s] = 765,003064375 Real Time[s] = 3 Completed 85%
Current Sim Time[s] = 774,023322262 Real Time[s] = 3 Completed 86%
Current Sim Time[s] = 783,044713021 Real Time[s] = 3 Completed 87%
Current Sim Time[s] = 792,000951318 Real Time[s] = 3 Completed 88%
Current Sim Time[s] = 801,010370516 Real Time[s] = 3 Completed 89%
Current Sim Time[s] = 810,019936587 Real Time[s] = 3 Completed 90%
Current Sim Time[s] = 819,007265386 Real Time[s] = 3 Completed 91%
Current Sim Time[s] = 828,017614770 Real Time[s] = 3 Completed 92%
Current Sim Time[s] = 837,008815110 Real Time[s] = 3 Completed 93%
Current Sim Time[s] = 846,006505914 Real Time[s] = 3 Completed 94%
Current Sim Time[s] = 855,030478574 Real Time[s] = 3 Completed 95%
Current Sim Time[s] = 864,052366162 Real Time[s] = 3 Completed 96%
Current Sim Time[s] = 873,064564290 Real Time[s] = 3 Completed 97%
Current Sim Time[s] = 882,072024056 Real Time[s] = 3 Completed 98%
Current Sim Time[s] = 891,081945395 Real Time[s] = 3 Completed 99%
Execution time : 4,0086 sec
Number of events (including timeouts) processed : 605409
Number of messages processed : 4
Number of context switches occurred : 12
Number of Local NULL messages sent : 0
Number of Remote NULL messages sent : 0
Total Number of NULL messages sent : 0
NULL messages / Regular messages : 0,000
[root@unimatrix-07]#
```

- کامپایلر جاوا باید بر روی سیستم نصب شده باشد. لازم به ذکر است که برای کامپایل برنامه (Java Runtime Environment) JRE کافی نیست و باید J2SDK (Java 2 Software Development Kit) را نیز از سایت [java.sun.com](http://java.sun.com) دانلود کنید.
- توضیح: به دلیل تحریم ایران از طرف سایت مذکور این فایل را از سایت دیگری دانلود کردیم.
- پکیج مربوط به لینوکس یک فایل از نوع rpm.bin است. پس از تعیین مجوز با دستور `chmod 755` می توانید فایل دانلود شده را اجرا نمایید. سپس با استفاده از `rpm -i` فایل RPM غیر فشرده را نصب کنید.
- کامپایلر جاوا `javac` و مفسر زمان اجرا `runtime java` در یک پوشه مانند `/usr/java/<version of java>/bin` قرار دارند. که توصیه می شود از نگارش ۱.۳ یا بالاتر استفاده کنید.

# نصب روی سیستم عامل ویندوز ایکس پی

- ابتدا باید برنامه های زیر در ویندوز نصب شده باشند
  - Visual Studio 6.0 w/ sp4
  - Java SDK 1.2+
- به هنگام نصب ویژوال استودیو زمانی که برای مقدار دهی متغیر های محیطی ( environment variables ) از شما سؤال می شود گزینه Accept را انتخاب کنید. اگر این گزینه را انتخاب نکرده اید می توانید با اجرای فرمان `vcvars.bat` متغیر ها را مقدار دهی کنید.
- ماشین مجازی جاوا را همانند مرحله قبل (منتها این بار برای سیستم عامل ویندوز) دانلود و نصب کنید.
- مسیر برنامه را به متغیر محیطی `PATH` اضافه نمایید.

با استفاده از برنامه winrar می توانید glomosim را در پوشه %programfiles% یا هر مسیر دلخواه (مثلا e:\glomosim) نصب کنید.



• متغیرهای محیطی مربوط به مسیر را مقدار دهی کنید. (مطابق شکل های صفحه بعد)

- **PCC\_DIRECTORY**

C:\Program Files\glomosim-2.03\parsec\windowsnt-4.0-vc6

- **PATH**

C:\Program Files\glomosim-2.03\parsec\windowsnt-4.0-vc6 \bin

- **Visual Studio vcvars.bat**

• مقدار متغیر محیطی `pcc` را از طریق فرمان `pcc -env` چک کنید.



## Environment Variables

### User variables for Admin

Variable	Value
include	C:\Program Files\Microsoft Visual Studio...
lib	C:\Program Files\Microsoft Visual Studio...
MSDevDir	C:\Program Files\Microsoft Visual Studio...
path	C:\Program Files\SUN\SDK\bin;C:\Progr...
PCC_DIRECTORY	E:\glomosim\parsec

New

Edit

Delete

### System variables

Variable	Value
ComSpec	C:\WINXP\system32\cmd.exe
FP_NO_HOST_C...	NO
NUMBER_OF_P...	2
OS	Windows_NT
Path	C:\WINXP\system32;C:\WINXP;C:\WIN...

New

Edit

Delete

OK

Cancel

C:\ Command Prompt

Microsoft Windows XP [Version 5.1.2600]  
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\auyeung>pcc -env

PCC\_DIRECTORY = c:\progra~1\glomosim-2.03\parsec\windowsnt-4.0-vc6

PCC\_CC = <none>

default: cl

PCC\_LINKER = <none>

default: cl -nologo -MT

PCC\_PP\_OPTIONS = <none>

default: -nologo -E -D\_\_STDC\_\_ -D\_POSIX\_ -D\_\_cdecl="" -D\_\_declspec=DECL

SPEC\_EATER

PCC\_CC\_OPTIONS = <none>

default: -nologo -MT -Za -c

PCC\_LINKER\_OPTIONS = <none>

default:

C:\Documents and Settings\auyeung>\_

• به دلیل اینکه ممکن است وجود فاصله در نام پوشه (Program Files) موجب بروز نقص در اجرای برنامه گردد احتیاطاً بهتر است که نام پوشه را بدون فاصله انتخاب نماییم. لذا به جای مسیر C:\Program Files\glomosim-2.03\parsec\windowsnt-4.0-vc6 مسیر E:\glomosim\parsec\windowsnt-4.0-vc6 (یا مشابه با این) را انتخاب می نماییم و PCC\_DIRECTORY را مقدار دهی می کنیم.

• مقادیر E:\glomosim\parsec\bin;E:\glomosim\parsec\runtime;

E:\glomosim\parsec\main; را به انتهای متغیر PATH اضافه می کنیم.

• مقدار E:\glomosim\parsec\runtime; را به انتهای متغیر محیطی lib می افزاییم.

• مقدار E:\glomosim\parsec\include را به انتهای متغیر محیطی include می افزاییم.

• در ضمن می توانیم تغییرات فوق را در یک فایل دسته ای (مثلا با نام `a.bat`) بنویسیم و قبل از اجرای `glomosim` یا ملحقات آن اجرا نماییم.

- `Set PCC_DIRECTORY= E:\glomosim\parsec\windowsnt-4.0-vc6`
- `Set PATH=%PATH%; E:\glomosim\parsec\bin;E:\glomosim\parsec\runtime; E:\glomosim\parsec\main;`
- `Set lib=%lib%; E:\glomosim\parsec\runtime;`
- `Set include=%include%; E:\glomosim\parsec\include;`

- برنامه makent.bat را از پوشه golosim\main اجرا کنید. پس از اتمام اجرای این برنامه، پوشه bin آماده اجرا خواهد بود.
- با فرمان javac \*.java داخل پوشه java\_gui فایل های جاوا را نیز کامپایل کنید.



```
Command Prompt - makent
oise.pc
tmp-788.c
tmp-788-0.c

>call pcc -0x -I..\include\ -I..\radio\ -clock longlong -c ..\radio\radio_nono
oise.pc
tmp-3376.c
tmp-3376-0.c

>call pcc -0x -I..\include\ -I..\mac\ -clock longlong -c ..\mac\802_11.pc
tmp-3440.c
tmp-3440-0.c

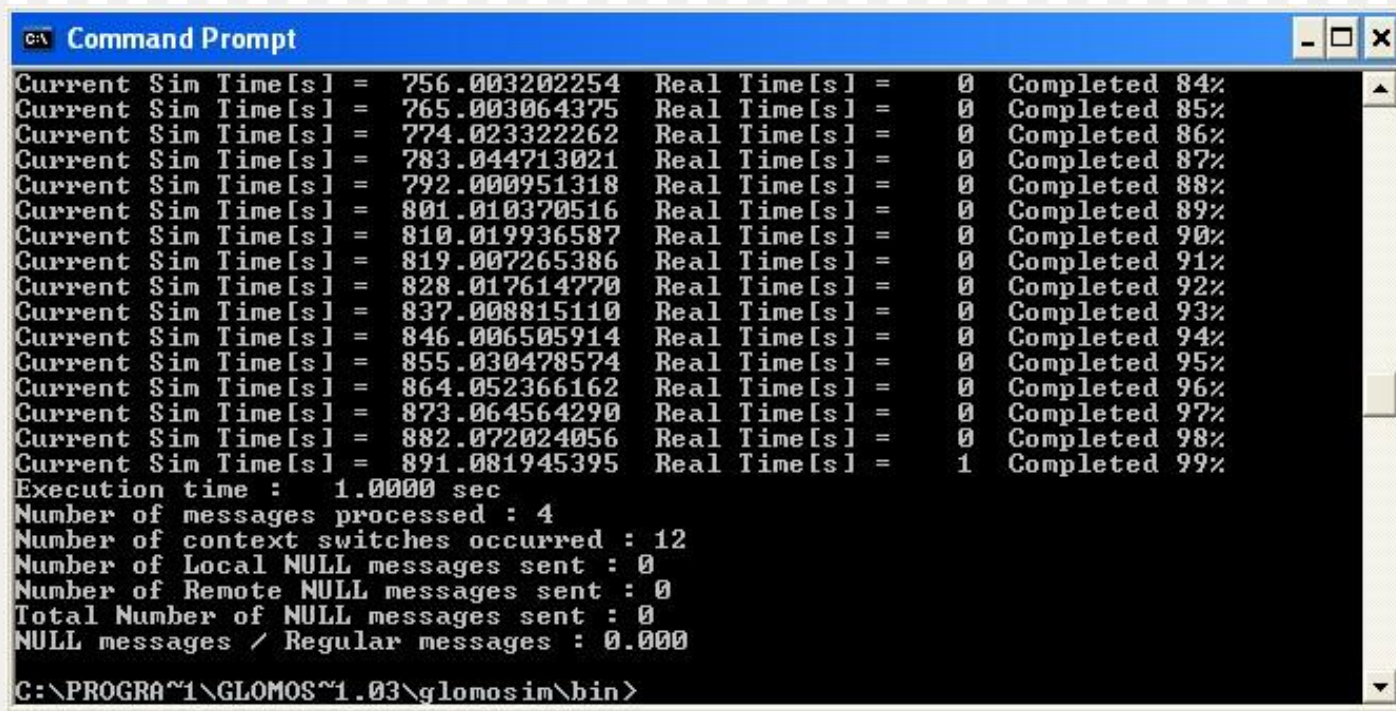
>call pcc -0x -I..\include\ -I..\mac\ -clock longlong -c ..\mac\csma.pc
tmp-3288.c
tmp-3288-0.c

>call pcc -0x -I..\include\ -I..\mac\ -clock longlong -c ..\mac\maca.pc
tmp-1132.c
tmp-1132-0.c

>call pcc -0x -I..\include\ -I..\mac\ -clock longlong -c ..\mac\tasma.pc
tmp-2080.c
tmp-2080-0.c
```

# اجرای برنامه در محیط ویندوز

- برای اجرای برنامه داخل پوشه bin فرمان glomosim datafile را وارد نمایید که به جای datafile نام فایلی را که داده ها در آن ذخیره شده است، بنویسید.



```
C:\ Command Prompt
Current Sim Time[s] = 756.003202254 Real Time[s] = 0 Completed 84%
Current Sim Time[s] = 765.003064375 Real Time[s] = 0 Completed 85%
Current Sim Time[s] = 774.023322262 Real Time[s] = 0 Completed 86%
Current Sim Time[s] = 783.044713021 Real Time[s] = 0 Completed 87%
Current Sim Time[s] = 792.000951318 Real Time[s] = 0 Completed 88%
Current Sim Time[s] = 801.010370516 Real Time[s] = 0 Completed 89%
Current Sim Time[s] = 810.019936587 Real Time[s] = 0 Completed 90%
Current Sim Time[s] = 819.007265386 Real Time[s] = 0 Completed 91%
Current Sim Time[s] = 828.017614770 Real Time[s] = 0 Completed 92%
Current Sim Time[s] = 837.008815110 Real Time[s] = 0 Completed 93%
Current Sim Time[s] = 846.006505914 Real Time[s] = 0 Completed 94%
Current Sim Time[s] = 855.030478574 Real Time[s] = 0 Completed 95%
Current Sim Time[s] = 864.052366162 Real Time[s] = 0 Completed 96%
Current Sim Time[s] = 873.064564290 Real Time[s] = 0 Completed 97%
Current Sim Time[s] = 882.072024056 Real Time[s] = 0 Completed 98%
Current Sim Time[s] = 891.081945395 Real Time[s] = 1 Completed 99%
Execution time : 1.0000 sec
Number of messages processed : 4
Number of context switches occurred : 12
Number of Local NULL messages sent : 0
Number of Remote NULL messages sent : 0
Total Number of NULL messages sent : 0
NULL messages / Regular messages : 0.000
C:\PROGRA~1\GLOMOS~1.03\glomosim\bin>
```

# اجرای برنامه در محیط ویندوز

- اگر می خواهید خروجی را نیز داخل یک فایل متنی داشته باشید می توانید با استفاده از piping این کار را انجام دهید.
- `E:\glomsim\glomsim\bin>glomsim.exe config.in > gs.out`  
دستور بالا داده ها را از فایل متنی `config.in` خوانده نتیجه اجرا را در `gs.out` ذخیره می نماید.
- مثال های موجود در پوشه `glomsim\glomsim\scenarios` را نیز می توانید داخل پوشه `bin` کپی کرده، اجرا نمایید.

# اجرای یک شبیه سازی پایه

- اجرای glomosim به محتویان فایل config.in بستگی دارد که به تعدادی از متغیرهای پایه ای اشاره می نمایم.
- به دلیل نام گذاری خوش تعریف، بیشتر متغیرها از روی نام شان قابل تشخیص اند و ما به مواردی که در config.in توضیح داده نشده اند، می پردازیم.



## SIMULATION-TIME •

- کل زمان شبیه سازی

## TERRAIN-DIMENSIONS •

- موقعیت محیط شبیه سازی

## NUMBER-OF-NODES •

- تعداد گره ها

- لازم به ذکر است که این ها پارامترهایی هستند که بیشتر به این کلاس نزدیک اند و radio pathloss و propogation بیشتر در حالت های سیار مورد مطالعه قرار می گیرند.

## NODE-PLACEMENT •

▪ روش جانمایی گره های شبکه

- random تصادفی
- grid جدولی
- uniform یکنواخت
- by input file allowed از طریق فایل ورودی

## MOBILITY •

▪ حرکت گره های شبکه

- none بدون حرکت
- random-waypoint مسیره های تصادفی
- by trace تعقیب
- by pathloss matrix ماتریس مسیر

## MAC-PROTOCOL •

■ تعريف نوع MAC استفاده شده:

CSMA •

TSMA •

MACA •

802.11 •

## ROUTING-PROTOCOL •

■ تعریف نوع پروتوکل مسیریابی

Bellman-Ford •

AODV •

DSR •

LAR •

WRP •

Fisheye •

ZRP •

Static •

• به نظر می رسد برخی پروتوکل ها هنوز پیاده سازی نشده اند. فایل های ZRP خالی هستند.

• انتخاب پارامترهای گزارش های آماری (به صورت yes/no)

- APPLICATION-STATISTICS (لایه کاربرد) ■
- TCP-STATISTICS ■
- UDP-STATISTICS ■
- ROUTING-STATISTICS (لایه مسیر یاب) ■
- NETWORK-LAYER-STATISTICS (لایه شبکه) ■
- MAC-LAYER-STATISTICS ■
- RADIO-LAYER-STATISTICS ■
- CHANNEL-LAYER-STATISTICS ■
- MOBILITY-STATISTICS ■
- GUI-OPTION ■
- GUI-RADIO ■
- GUI-ROUTING ■

## • سایر فایل های مورد استفاده

■ **glomo.stat** حین شبیه سازی ایجاد می گردد و شامل کلیه آمار ها است. از طرفی به دلیل این که به تفکیک لایه و گره ایجاد گردیده، محتویات آن به آسانی قابل فیلتر شدن است.

■ **mcast.conf** شامل اطلاعات مالتی کست می باشد.

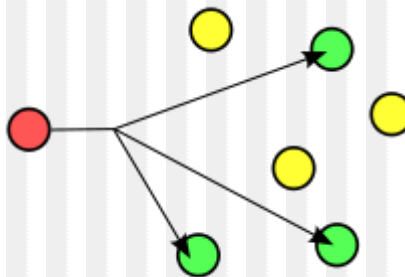
• ستون اول آدرس گره مبداء `source-node-address`

• ستون دوم آدرس مالتی کست `multicast address`

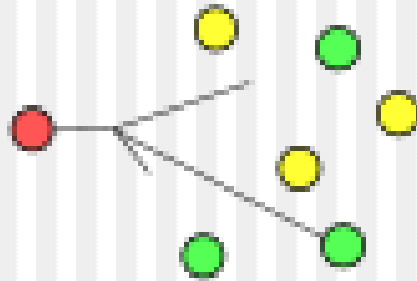
• ستون سوم زمان اتصال `join time`

• ستون چهارم زمان انفصال `leave time`

- Multicast آدرس دهی مالتی کست، یک فناوری در شبکه است که اطلاعات را به طور همزمان به گروهی از گره های مقصد تحویل می دهد و از این راهکار استفاده می کند که هریک از پیوند های شبکه، پیام را فقط یک بار تحویل می دهد و اطلاعات را فقط وقتی کپی می کند که پیوند ها به دو شاخه تقسیم شوند و به دو مقصد بروند.

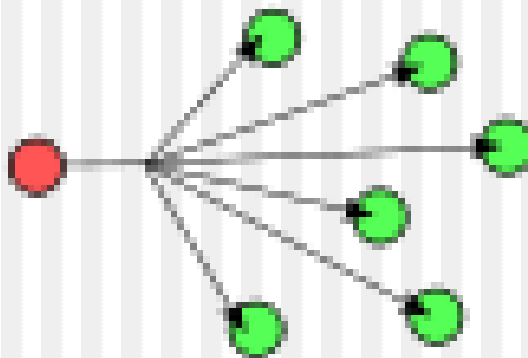


- **anycast** انیکست نوعی آدرس دهی و مسیریابی در شبکه است که داده در آن به مقصدی هدایت می شود که از دیدگاه همبندی مسیریابی، نزدیک ترین و بهترین است.

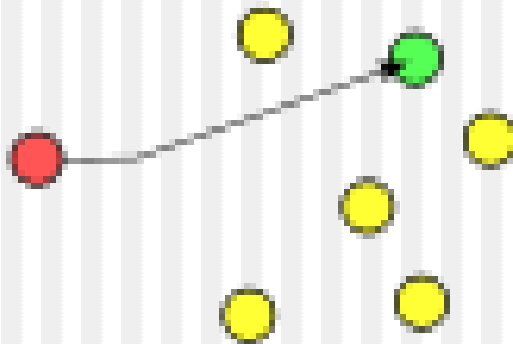




- Broadcasting در شبکه رایانه‌ای، انتشار به انتقال بسته اطلاعاتی که به وسیله دستگاهی در شبکه دریافت می شود اطلاق می گردد.



- Unicast در شبکه رایانه ای انتقال یونیکست انتقال بسته‌های اطلاعاتی است به یک رایانه مقصد (نه بیشتر)



# مشاهده نتایج در GloMoSim VT

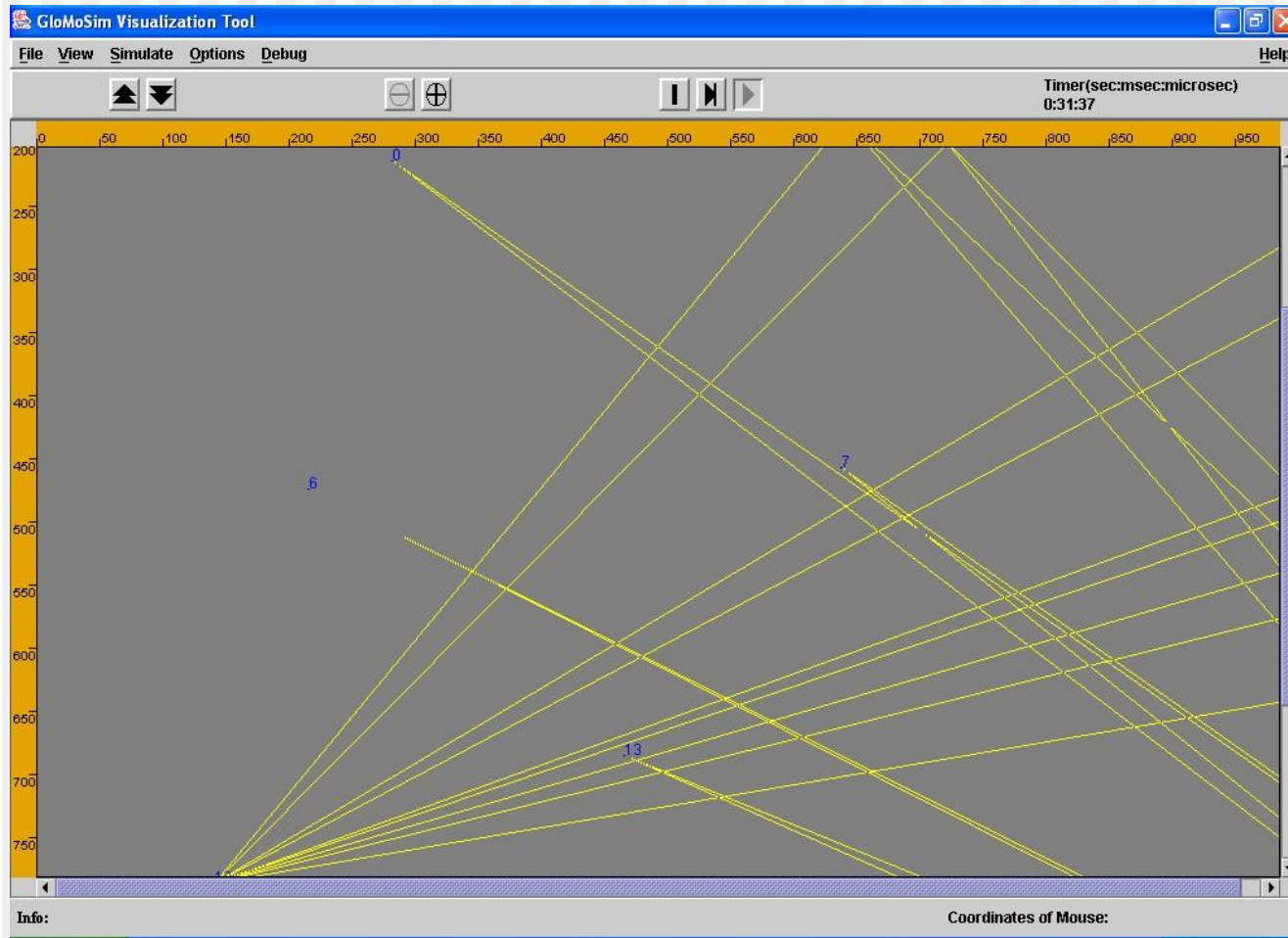
- ابزار تصویری (Visualization Tool) این امکان را به شما می دهد که به صورت دیاگرام های گرافیکی نتیجه شبیه سازی را مشاهده نمایید.
- برای استفاده از این ابزار
  - Glomosim را داخل VT اجرا نمایید.
  - یک trace file ذخیره کنید
  - فایل ایجاد شده را پخش (Play) کنید.

# نمونه اجرا به صورت تصویر متحرک

برای شروع نمایش، روی تصویر کلیک کنید.



# نمایی از اجرا در GloMoSim VT



# پروتکل های دسترسی چند گانه با قابلیت شنود سیگنال CSMA

به مجموعه این پروتکل ها، پروتکل های شنود سیگنال نیز گوییم. این پروتکل ها مناسب تر بوده و کاربردی تر هستند. چراکه بدون تحقیق حکم صادر نمی کنند!

## Persistent CSMA

به این پروتکل که گاهی به آن ۱- Persistent CSMA نیز گوییم طبق الگوریتم زیر کار می کند:

۱. به کانال گوش بده.

۲. اگر کانال خالی است اطلاعات را به سوی کانال ارسال کن.

۱. در غیر این صورت به مرحله ۱ بازگرد.

در این روش نیز امکان تصادم وجود دارد. دو حالت زیر سناریوهای احتمالی وقوع این رویداد است:

- تاخیر انتشار وجود داشته باشد.
- انتظار همزمان برای خالی شدن کانال

## Nonpersistent CSMA

این پروتکل مبتنی بر شنود سیگنال است و طبق الگوریتم زیر است:

۱. به کانال گوش بده.
  ۲. اگر کانال خالی است اطلاعات را به سوی کانال ارسال کن.
    ۱. در غیر اینصورت زمان تصادفی را صبر کن.
  ۳. بازگشت به مرحله ۱.
- این روش با حالت قبل دو تفاوت دارد:
- تاخیر بیشتری نسبت به روش قبل دارد.
  - زمان انتظار به صورت تصادفی محاسبه می گردد.



## p-Persistent CSMA

این پروتکل برای کانالهای زمان بندی مناسب است و دارای الگوریتم زیر است:

۱. شنود کانال.

۲. اگر کانال اشغال نیست آنرا با احتمال  $p$  به روی کانال قرار بده (این امر قطعی نیست).

۱. در غیر این صورت تا بعد از اسلات بعدی صبر کن و به مرحله ۱ بازگرد.

تا اینجا چند پروتکل را بررسی کردیم، قبل از ادامه این بخش توجه شما را به شکل روبرو جلب می کنیم که به مقایسه بهره وری کانال (ظرفیت مفید) بر حسب بار پرداخته است. قبل از بحث در مورد ادامه پروتکل ها این نکته را خاطر نشان می کنیم که تمام پروتکل هایی که تا اینجا بررسی کردیم دسترسی تصادفی را به کانال دارند.

## CSMA با تشخیص تصادم

در این پروتکل به محض تشخیص تصادم ادامه ارسال فریم آسیب دیده متوقف می گردد. این قطع شدن سریع، باعث می شود که در پهنای باند به میزان زیادی سرفه جویی شود. چنین پروتکلی را به اختصار **CSMA/CD** یا **پروتکل دسترسی چندگانه با قابلیت شنود سیگنال حامل همراه با تشخیص تصادم** نامیم. CSMA/CD می تواند در سه وضعیت باشد:

- رقابت
- ارسال
- بیکار

مرجع: شبکه های کامپیوتری - آندرواس. تنن بام - ویراست چهارم ۲۰۰۳

برای ادامه به صفحه ۲۷ بازگردید.