
1. Download bochs and pintos

- 1. download bochs-2.6.7.tar.gz
- 2. download pintos.tar.gz

2. Use the tar command to decompress two files separately

```
tar -zxvf bochs-2.6.7.tar.gz
tar -zxvf pintos.tar.gz
```

3. Install the following software

```
sudo apt-get install build-essential
sudo apt-get install xorg-dev
sudo apt-get install bison
sudo apt-get install libgtk2.0-dev
```

4. Install bochs

```
cd bochs-2.6.7
./configure --with-nogui --enable-gdb-stub
make
sudo make install
```

test bochs

```
bochs --version
```

```
=====
Bochs x86 Emulator 2.6.7
Built from SVN snapshot on November 2, 2014
Compiled on Mar 13 2018 at 17:35:28
=====
Usage: bochs [flags] [bochsrc options]

-n          no configuration file
-f configfile  specify configuration file
-q          quick start (skip configuration interface)
-benchmark N  run bochs in benchmark mode for N millions of emulated ticks
-dumpstats N  dump bochs stats every N millions of emulated ticks
-r path       restore the Bochs state from path
-log filename specify Bochs log file name
--help       display this help and exit
--help features  display available features / devices and exit
--help cpu    display supported CPU models and exit

For information on Bochs configuration file arguments, see the
bochsrc section in the user documentation or the man page of bochsrc.
00000000000p[      ] >>PANIC<< command line arg '--version' was not understood
=====
Bochs is exiting with the following message:
[      ] command line arg '--version' was not understood
=====
00000000000i[SIM  ] quit_sim called with exit code 1
=====
```

5. Running pintos on bochs

```
cd pintos/src/threads
make
cd build
../../utils/pintos --run alarm-multiple
```

```
squish-pty bochs -q
=====
Bochs x86 Emulator 2.6.7
Built from SVN snapshot on November 2, 2014
Compiled on Mar 13 2018 at 17:35:28
=====
000000000000i[      ] reading configuration from bochsrc.txt
000000000000e[      ] bochsrc.txt:8: 'user_shortcut' will be replaced by new 'keyboard' option.
000000000000i[      ] installing nogui module as the Bochs GUI
000000000000i[      ] using log file bochsout.txt
Pilo hda1
Loading.....
Kernel command line: run alarm-multiple
Pintos booting with 4,096 kB RAM...
383 pages available in kernel pool.
383 pages available in user pool.
Calibrating timer... 204,600 loops/s.
Boot complete.
Executing 'alarm-multiple':
(alarm-multiple) begin
(alarm-multiple) Creating 5 threads to sleep 7 times each.
(alarm-multiple) Thread 0 sleeps 10 ticks each time,
(alarm-multiple) thread 1 sleeps 20 ticks each time, and so on.
(alarm-multiple) If successful, product of iteration count and
(alarm-multiple) sleep duration will appear in nondescending order.
(alarm-multiple) thread 0: duration=10, iteration=1, product=10
(alarm-multiple) thread 0: duration=10, iteration=2, product=20
(alarm-multiple) thread 1: duration=20, iteration=1, product=20
(alarm-multiple) thread 2: duration=30, iteration=1, product=30
(alarm-multiple) thread 0: duration=10, iteration=3, product=30
(alarm-multiple) thread 0: duration=10, iteration=4, product=40
(alarm-multiple) thread 1: duration=20, iteration=2, product=40
(alarm-multiple) thread 3: duration=40, iteration=1, product=40
(alarm-multiple) thread 4: duration=50, iteration=1, product=50
(alarm-multiple) thread 0: duration=10, iteration=5, product=50
```