

CENG 311 – Lab 10 Summary

Important Notes

- I strongly recommend installing the Pin tool and practicing before the following lab session if you have a personal computer. There are the Pin tools in some of the lab computers, but I don't know about all of them.
- If you have a question or problem, do not wait for the lab session; ask online.
- It seems there is no problem with using the Pin tool on AMD processors, at least for studying. You may use this tool even if you have no Intel processor.
- If you use macOS, and if you cannot run the Pin tool, please look at this sentence:

"Note that if SIP (System integrity Protection) is an enabled on the machine then Pin will not be able to run system files. The only way to run system files will be to disable SIP on that machine. More information about SIP and how to disable it can be found online. Some runtime functions may run indirectly system binaries which are SIP protected that Pin cannot run. Example for it is system() function which calls /bin/sh which is SIP protected."

Reference: Pin 3.21 User Guide, <https://software.intel.com/sites/landingpage/pintool/docs/98484/Pin/html/>, Last Accessed: 26 December 2023

Checking the Processor Information

There are multiple ways to check the processor information. There are some methods to do this below if you use a Linux distro.

- Ubuntu uses the GNOME Desktop Environment as default. You can find the used processor using this path: "Settings → About → Processor". Even if you use another desktop environment such as KDE or XFCE, there must be some alternative way to check it.
- `neofetch` is a practical package for checking some information, including the title of the processor. You can install this package like that: "`$ sudo apt install neofetch`", for the Debian-based distros such as Ubuntu. Then, you can run this package by writing "`$ neofetch`" into the terminal.
- `lscpu` is a practical package for checking some information, including the title of the processor. You can install this package like that: "`$ sudo apt install util-linux`", for the Debian-based distros such as Ubuntu. Then, you can run this package by writing "`$ lscpu`" into the terminal.
- `cpu-x` is a package with a graphical user interface for checking detailed information about the processor used. You can install this package like that: "`$ sudo apt install cpu-x`", for the Debian-based distros such as Ubuntu. Then, you can run this package by writing "`$ cpu-x`" into the terminal or finding its icon in the "Show Applications" tab.

If you use macOS, you can try the packages mentioned above. Look at the brew package manager; these packages or their alternatives must exist. If you use Windows, you can press the "Win + R" combination, write `dxdiag` into the textbox pop, and confirm the pop-up. Some information about your processor, graphic card, RAM, etc., exists with a GUI.

Installing the Pin Tool

Installing the Pin tool is straightforward. Just use [this link](#) (Last Accessed: 26 December 2023), select the proper link, and download. You can extract the compressed file wherever you want. You can work with the Pin tool in the directory you extracted.

Working with the Pin Tool

Firstly, compile a code and move the executable binary file into the Pin directory. Then, you need to set the profiler.

```
$ set INTEL_JIT_PROFILER intel64/lib/libpinjitprofiling.so
```

You need to repeat this for each terminal session. If you want to write this for each session, you can append that command to the end of the `~/.bashrc` for Linux distros and if you use Bash as the shell.

You need to make the tools that you want to use. So, go to the directory in which the tools are included. Make sure you are in the Pin directory.

```
$ cd source/tools/ManualExamples
```

If you want to count the instruction, you can make `inscount0.test`. If you want to trace instructions as hexadecimal machine codes, you can make `itrace.test`.

```
$ make inscount0.test
```

```
$ make itrace.test
```

Now, you can return to the main Pin directory.

```
$ cd ../../..
```

To run the Pin tool for counting the number of the instructions:

```
$ ./pin -t source/tools/ManualExamples/obj-intel64/inscount0.so -- ./executable_you_moved
```

To run the Pin tool for tracing the instructions:

```
$ ./pin -t source/tools/ManualExamples/obj-intel64/itrace.so -- ./executable_you_moved
```

The result of the inscount0 is in the inscount.out, and the result of the itrace is in the itrace.out.

```
$ cat inscount.out
```

```
$ cat itrace.out
```

Document Information

- **Lab Dates:** 21-22 December 2023
- **Last Edited:** 26 December 2023
- **Version:** 1.0.0
- **Author:** Nuri Furkan Pala
- **E-mail:** nuripala@iyte.edu.tr