**READ ME**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

This tool plugs into your eclipse workbench as a separate header “Pin”.

Features: Works only in Linux. Applicable for multi-threaded C/C++ programs written using pthreads libraries.

Prerequisites:

1. You must have jdk installed in your system. Type **sudo apt-get install default-jdk** on your terminal.
2. You must have graphviz (Visualization software) installed in your system. Type **sudo apt-get install graphviz** on your terminal.
3. Most of the linux installations come pre installed with **xterm**. In case you have uninstalled it you need to reinstall the same. You can use the command **sudo apt-get install xterm** from the terminal.
4. 8 GB of RAM atleast

**The Step by Step Process to get this tool working in your system: The first step is primarily based on downloading the multiple zipped files and placing them in correct directories. This step is a bit longer due to the github size limit of 100MB for file upload, hence we had to break down our files.**

* There are three zipped folders:- **EclipseP, pin-3.0-76991-gcc-linux** and **PinPlugin and a folder “plugins”**  Download the folder and all of these files, unzip it in your local file system and remember the path. Now do the following move operations

Move the unzipped **pin-3.0-76991-gcc-linux** into **unzipped EclipseP**

Move the folder **plugins** into **unzipped EclipseP**

The folder PinPlugin will not be moved , hence it will be a separate folder, hence now you have two main folders **EclipseP and PinPlugin**

* Then cd (change directories) to ***Path to EclipseP/ EclipseP*** and then type **./eclipse**
* The eclipse IDE launches. Go to File and create a new java project by selecting a workspace.
* Now right click on the new project created and click on Import
* Go to ***General/FileSystem/Path to folder PinPlugin/PinPlugin*** that is import the whole folder PinPlugin from the directory where it was extracted. While importing, allow overwriting of the existing .project files.
* In few cases, IF and ONLY IF errors flash in the below error screen after importing, in such a case follow the below steps to quick fix the issue.

The screenshot is given below (8 errors)



Expand the imported folder and right click on JRE System Library.

Go to Build Path and then to Configure Build Path.

A screen opens, select all the missing jar files above the folder JRE System Library (highlighted in red in the screenshot)

And click on **Remove. (Note:** If you scroll and read these jars highlighted in red, you will see these are displayed as missing)



Then Click on folder EclipseP folder (present at the right of the screen) and then go to Plugins/BuildPathFixPlugins and

Select all the files of the folder and then click OK.

The error will be removed. Screenshot given below



* Click on plugin.xml-> “Overview” tab-> Launch an Eclipse application.

Screenshot attached



* In the newly opened Eclipse application, change the perspective to C/C++. You will find a “**PinTool**” option in the topmost menu bar where you have file, edit, view etc. Load any .c/.cpp file, debug and run it and then try the different sub options of the menu Pintools for data race detection (with and without debugger), deadlock detection, and decision tree on different thread models. Select any suboption and click on them to run the tool. Graphical outputs will pop up and the textual output will be shown in a new terminal.

In case of thread models, you will be prompted to select the model for which you want to see the graphical traces. Also all the graphical and textual output files will be available at path (**path to EclipseP/EclipseP/pin-3.0-76991-gcc-linux/sources/tools/ManualExamples**) after every execution.

Some screenshots given





* **Note:** In some cases it might happen that you need to give some permission to the scripts involved in running the tools. In that case please go to the properties of the scripts involved in the main **eclipse folder** and also inside the

“**path to EclipseP/EclipseP/pin-3.0-76991-gcc-linux/sources/tools/ManualExamples**” folder and give **permission to be executed** as a program to all the “**.sh**” files.