Exercise 2:

Write a Pintool (in JIT mode) that prints into a file called "**loop-count.csv**" the profiling about executed loops in each routine (RTN).

No need to handle loops that are implemented using indirect jumps.

The pintool should be named "**ex2.so"**.

For each loop with a non-zero **CountSeen**, the tool should emit the following information, in this exact format:

0x <loop$_1$ address>, <loop$_1$ CountSeen>, <loop$_1$ CountLoopInvoked>, <loop$_1$ MeanTaken>, <loop$_1$ DiffCount>, <loop$_1$ routine name>, 0x <loop$_1$ routine address> , <loop$_1$ routine instructions count>

0x <loop$_2$ address>, <loop$_2$ CountSeen>, <loop$_2$ CountLoopInvoked>, <loop$_2$ MeanTaken>, <loop$_2$ DiffCount>, <loop$_2$ routine name>, 0x <loop$_2$ routine address> , <loop$_2$ routine instructions count>

…

0x <loop$_n$ address>, <loop$_n$ CountSeen>, <loop$_n$ CountLoopInvoked>, <loop$_n$ MeanTaken>, <loop$_n$ DiffCount>, <loop$_n$ routine name>, 0x <loop$_n$ routine address> , <loop$_n$ routine invocations count>

Where:

CountSeen = total number of times the loop's backward edge was executed

CountLoopInvoked = number of times the loop was invoked

MeanTaken = average number of iterations taken for the loop invocations

DiffCount = number of times that two successive loop invocations took a different number of iterations

routine name/address/ = Routine name/address in which the loop resides and the number of times it was called.

routine invocations count = Number of time the routine that contains the loop was called/invoked.

The above loops' list should be ordered according to highest **CountSeen** down to lowest **CountSeen**.

You can assume that the total number of loops is no larger than 10,000 and number of total routines no larger than 1000.

The pintool should note run longer than 1 second (elapsed time) on the bzip2 input.

Test your pintool:

In the moodle you'll find the input binary file called "**bzip2.gz**" along with an input file to give it called "**input.txt.gz.**
Ftp the files to your T2 Linux account and open them using the **gunzip** command.
To run it simply type: $ **./bzip2 –k –f input.txt**
This will compress the file **input.txt** and generate a new file **input.txt.bz2**

To test your pintool on the above **bzip2** binary file, simply type:
<pindir>/pin –t ex2.so -- ./bzip2 –k –f input.txt

Submission requirements:
The submission of this exercise is **in pairs only**.

Submit 1 compressed file called **“ex2.zip”** into the moodle exercise2 link containing the following files:

1. The binary of your pintool **ex2.so** (compiled, and tested by you that it runs and gives the result).
2. A directory called: ‘**src**’ containing all the sources of your pintool along with a **REDAME.txt** file that describes the compilation command and how to run the tool.

Submission deadline: midnight Sunday May 13, 2018.