

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 <loop₁ target address>, <loop₁ CountSeen>, <loop₁ CountLoopInvoked>, <loop₁ MeanTaken>, <loop₁ DiffCount>, <loop₁ routine name>, 0x <loop₁ routine address> , <loop₁ routine instructions count>

0x <loop₂ target address>, <loop₂ CountSeen>, <loop₂ CountLoopInvoked>, <loop₂ MeanTaken>, <loop₂ DiffCount>, <loop₂ routine name>, 0x <loop₂ routine address> , <loop₂ routine instructions count>

...

0x <loop_n target 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 not 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 12, 2019.