Project:

Write a Pintool in JIT mode and in Probe mode.

The pintool receives 2 possible knobs: "**-prof**" and "**-opt**" that are to be applied in 2 stages as follows:

1. **<pindir>/pin –t project.so –prof -- <run command>**
2. **<pindir>/pin –t project.so –opt -- <run command>**

When applied with the "**-prof**" knob, the pintool should preferably count executed instructions in the **MainExecutable** image only and print them into the file "**count.csv**" in any desired format.

Feel free to modify and set the format of "**count.csv**" as you see fit, in order to collect any profiling information needed for your optimization with the "**-opt**" option.

When applied with the "**-opt**" knob, the pintool should run in probe mode and apply **Function Inlining** of functions that have a single Hot call site, followed by **Code Reordering**.

For the project use the provided pintool source code "**rtn-translation.cpp**" located at:

https://moodle2223.technion.ac.il/mod/resource/view.php?id=168217

As well as the pintool for reverting conditional jumps at:

https://moodle2223.technion.ac.il/mod/resource/view.php?id=168226

Run your pintool:

In the moodle you will find the input binary files called "**bzip2.gz**", **"mcf.gz"** and "**cc1.gz**" along with their corresponding input files **"input-long.txt", "inp.in"** and **"200.i"** for both the profile and the optimization stages. Ftp the files to your Linux account and open them using the **gunzip** command.

For the -**opt** option the pintool should show a reduction of at least **15%** for **at least one** of the performance metrics as measured by the "perf stat" Linux command below:

- Total number of branches
- Total number of i-cache misses
- Total I-TLB misses
- Total number of page faults
- Number of Context Switches
- User time or Elapsed time

How to measure performance:

To measure your pintool performance gain on the above binary files, use the **-no_tc_commit** option to disable the "commit" function in **rtn_translation.cpp** and compare the runs (both via pin) between the non-commit pintool and the pintool that enables the commit for the optimized code (Inline followed by Code Reordering):
Use the "**perf stat**" command twice as follows:

- bzip2:
  1. **perf stat <pindir>/pin -t project.so –opt <max-loop-unroll-num> -- ./bzip2 –k -f input-long.txt**
  2. **perf stat <pindir>/pin -t project.so –opt <max-loop-unroll-num> -no_tc_commit -- ./bzip2 –k –f input-long.txt**
- cc1:
  1. **perf stat <pindir>/pin -t project.so –opt <max-loop-unroll-num> -- ./cc1 200.i -o 200.s**
  2. **perf stat <pindir>/pin -t project.so –opt <max-loop-unroll-num> -no_tc_commit -- ./cc1 200.i -o 200.s**
- mcf:
  1. **perf stat <pindir>/pin -t project.so –opt <max-loop-unroll-num> -- ./mcf inp.in**

2. **perf stat <pindir>/pin -t project.so –opt <max-loop-unroll-num> -no_tc_commit -- ./mcf inp.in**

The first run should show improvement in one of the perf stat statistics.

Note: "perf stat" does not provide hardware statistics on a virtual machine and requires to install a Linux machine in a dual boot mode.

Submission requirements:

The submission of this exercise is **in pairs only**.

Submit 1 compressed file called **"project.zip"** into the moodle project link containing the following files:

1. The binary of your pintool **project.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 the make files and a REDAME.txt file that includes the following:
    a. names + id numbers
    b. compilation command
    c. how to run the tool.
    d. Explain the format of the profile file **count.csv**
    e. Explain the pattern of the loop candidates which you were using to unroll.
    f. Describe the threshold used to distinguish between frequently and rarely executed instructions.

Submission deadline: midnight Sunday September 7, 2023.

Project defense:

On September 7th, we will publish several dates and time slots for each pair to choose form, in order to defend their project.

There is an option to defend the project via zoom for students in abroad.

During defense, each pair must present foils describing their solution along with measurement results.

Foils need to include a presentable walk thru on the code implementation of the optimizations.

Good Luck!