**Final Project:**

Write a Pintool in probe mode that identifies illegal memory overflow accesses in a running program.

In order to implement the pintool use the provided 2 pintools in the moodle project [link](https://moodle.technion.ac.il/mod/assign/view.php?id=469033) **rtn-translation-mt.cpp** and **my\_malloc\_trace.cpp** along with your pintool from exercise 4.

**Test your pintool:**

The project should be applied on 2 given binaries as follows:

1. **test\_app** – apply the pintool on the provided **test\_app** executable in the moodle project [link](https://moodle.technion.ac.il/mod/assign/view.php?id=469033) as follows:  
   **<pindir>/pin -t project.so -- ./test\_app**make sure the instruction that performs the illegal memory access is detected and printed out to the screen along with its address at runtime.
2. **bzip2 -** apply the pintool on **bzip2** for checking general functional correctness as follows: **<pindir>/pin -t project.so -- ./bzip2 -k -f input.txt**make sure that the output file **input.txt.bz2** is identical to the same file generated by original bzip2 using the diff command and verify it returns without any messages.

**Submission requirements:**

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

Submit 1 compressed file called **“project.zip”** into the moodle project [link](https://moodle.technion.ac.il/mod/assign/view.php?id=469033) 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 a **REDAME.txt** file that describes the compilation command and how to run the tool.

**Determining the project’s grade:**

1. The final project grade will include the code itself along with an oral defense meeting.
2. Base grade is 85.
3. To reach a perfect grade, the pintool must detect the illegally memory accessing instruction with minimal performance degradation on **bzip2** of no more than 10% compared to the original run without the pintool.  
   To measure the pintool performance use “**time”** (or “**perf stat”)** command as follows:

**$ time <pindir>/pin -t project.so -- ./bzip2 -k -f input-long.txt**

**$ time ./bzip2 -k -f input-long.txt**

Make sure to run each command 3 times and check for stable results.

To reach the performance goal try to apply some optimizations on the translated and instrumented code as you have learned during the course.  
Tip: Consider adding probe jumps not only in heads of routines in order to reduce the overhead of the translated code.

**Submission deadline: midnight of August 31st, 2017.**