```
################################################################################
# CS:APP Cache Lab
# Directions to Instructors
#
# Copyright (c) 2013, R. Bryant and D. O'Hallaron, All rights reserved.
################################################################################
```

This directory contains the files that you will need to run the CS:APP
cache lab, which develops the student's understanding of caches.

****
1. Overview
****

In this lab, the student works on two C files called csim.c and
trans.c. There are two parts: Part (a) involves implementing a cache
simulator in csim.c. Part (b) involves writing a function that
computes the transpose of a given matrix in trans.c, with the goal of
minimizing the number misses on a simulated cache.

Each time a student with login "foo" compiles their work, the Makefile
automatically generates a handin file, called foo-handin.tar, that
contains the csim.c and trans.c file. Students hand this tar file in
to the instructor.

The driver program (driver.py) evaluates the correctness of the cache
simulator in csim.c, and the performance and correctness of the
transpose functions in trans.c. See the writeup for details.

Requirements:
- The lab must be done on a 64-bit x86-64 system.
- The driver requires a version of Valgrind (http://valgrind.org) that
supports the "--tool=lackey" option.

****
2. Files
****

```
Makefile            Builds the entire lab
README              This file
grade/              Autograding scripts for the instructor
src/                Source code of this lab
cachelab-handout/   Handout directory that goes to the students. This directory
                    is completely generated from files in the src directory
                    Never put any hard state in this directory.
cachelab-handout.tar  Tar file of handout directory that goes out to the students.
writeup/            Latex writeup about the lab. Modify to reflect your
                    environment.
```

****
3. Building the Lab
****

To build the default version of the lab, modify the Latex lab writeup in ./writeup/cachelab.tex for your environment. Then type the following in the current directory:

```
unix> make clean
unix> make
```

This will build the cachelab-handout/ directory and its cachelab-handout.tar archive that you can handout to students. The command:

```
unix> make dist DEST=<DIR>
```

will copy the tarfile and copies of the writeup to directory <DIR>, where the students can access it.

****
4. Handing in the Lab
****

Each time a student with login "foo" compiles their work, the Makefile automatically generates a handin file called foo-handin.tar. If you want to autograde the handins, collect all of the student handin tar files in the ./grade/handin directory.

****
5. Autograding the Lab
****

You can run the driver program manually on each student's handin, or you can use the autograding scripts in ./grade to automatically grade each of the .tar files in the handin/ directory.

See ./grade/README for instructions.