```
########################################################################
# CS:APP Performance Lab
# Directions to Instructors
#
# Copyright (c) 2002, R. Bryant and D. O'Hallaron, All rights
reserved.
# May not be used, modified, or copied without permission.
#
########################################################################
#
```

This directory contains the files that you will need to run the CS:APP
Performance Lab, which develops the students's understanding of the
performance impact of caches and code optimizations such as blocking
and loop unrolling.

```
**********
1. Overview
**********
```

In this lab, students work on a C file, called kernels.c, that
contains some image processing kernels. The students are asked to use
the concepts they have learned in class to make these kernels run as
fast as possible.

Students evaluate the performance of their kernels by linking
kernels.c into a driver program called driver.c. Students can
register
different versions of their kernel functions with the driver. When
the
driver runs, it will execute, measure, and report on the performance
of each registered function. This is a very useful versioning feature
that allows students try out different approaches and do side by side
comparisons.

You should strongly encourage your students to take advantage of this
versioning feature in the driver.

```
********
2. Files
********
```

```
Makefile                Builds the entire lab
README                  This file
src/                    Driver code that runs and measures solutions
perflab-handout/        Handout directory that goes to the students
grade/                  Autograder code
writeup/                Sample Latex lab writeup
```

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

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

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

The Makefile generates the driver code, formats the lab writeup, and
then copies the driver code to the perflab-handout directory.
Finally, it builds a tarfile of the perflab-handout directory (in
perflab-handout.tar) which you can distribute 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. Autograding the Lab
****

There is an autograding script that automatically grades the lab.
See
./grade/README for instructions.