CIS 330: Project #3H
Assigned: May 30th, 2015
Due "June 5th"
NOTE: no late penalty will be applied to this project. That said, it must be submitted before June 12th.

Worth 11% of your grade

Please read this entire prompt!

You will be running a set of regression tests designed by previous students in this course. Every regression test has its own directory. The directory has source code called "stress_test.C" which contains their pipeline. It also has the correct output. If the program should throw an exception, then there is a file in the test's directory called "exception" with the text "Exception found!". If the program should generate an image, then the file in their directory called "checksum" with the correct checksum.

Directions:

(1) Download Proj3H.tar from the website.
   a. tar xvf Proj3H.tar
   b. cd 3H
   c. copy your source code into the sandbox directory

(2) run the tests and confirm you get the same answer
   a. run_all # runs all tests, declares passes and fails
   b. run_one # runs a single test … used to debug a mismatch
   c. run_one_w_image # runs a single test & creates its output image
   d. Note: not all tests submitted were valid … you should only worry about the tests on "valid_list". "run_all" automatically takes this list into account.

What to submit:
(1) a tarball of your final source code (via BlackBoard)
(2) a screenshot of you calling "run_all" and it stating how many valid tests you have

Scoring rubric:
0 incorrect: 11 points
1-5 incorrect: 8 points
6-10 incorrect: 7 points
11-half correct: 6 points
< half correct: 5 points

== What to do if your code doesn't match up? ==

You ran a test and feel like your program is running right? The first step is to run "run_one_w_image", look at the output. Then look at my baseline image at http://ix.cs.uoregon.edu/~hank/330/projects/3H/<testname>.png.

Note: test "jamesl" is a tough one. Debugging that will require some thinking and some new code. I expect that most solutions to that program will be hack-y. Not "if jamesl then throw exception hacky", but probably involve some global variables or something. Also note that if you pass that test, I intend to look at your code and see what you did. If it is super hacky, I'll likely mark the test as incorrect.