

Program 1 Test  
CSI4337

Compile with `gcc -Wall -std=gnull -g -lm triangles.c` and test on `and.ecs.baylor.edu`

Run with: `valgrind --tool=memcheck --show-reachable=yes --leak-check=full --track-fds=yes ./<executable>`

-4 points for leaked file descriptors up to -10 (ignore `fd=0`, 1, and 2)

-5 points for each memory foul/leak up to -20. VERIFY submission is responsible for leak

-2 (max -5) for each compiler warning (do not penalize for “suggestion warnings” like adding parentheses)

-8 bad parameter usage or test (test with 0 and 3 parameters)

Verify use of multiple processes (see `fork()`).

**Test 1 (15 points)**

Run with `test_spec_list.txt` and 3 processes with `valgrind`. Look for any memory leaks/violations, `fd` leaks, etc. Expect 4

**Test 2 (20 points)**

Run with `test_long_list.txt` and 10 processes. Expect 32909. Measure performance with `/usr/bin/time`. Report total time.

max 10 minutes; naïve serial on ECS Linux: 7:13.66 secs

**Test 3 (15 points)**

Run with `test_time_list.txt` and 4/8 processes. Expect 2161. Measure performance for both 4 and 8 with `/usr/bin/time`.

When increasing the processes from 4 to 8, the execution time should be cut by no less than 40%.

max 3 minutes; naïve serial on ECS Linux: 27.7 secs

**Test 4 (5 points)**

Run with `test_no_perm.txt` (file with no permission to read) and 2 processes. Expect graceful termination with useful error message PLUS “Permission denied” (and no number of triangles).

**Test 5 (5 points)**

Run with `test_bad_format.txt` and 2 processes. Expect graceful termination with helpful error message (and no number of triangles).

**Test 6 (10 points)**

Run with `test_too_short.txt` and 2 processes. Expect graceful termination with helpful error message (and no number of triangles).

**Test 7 (5 points)**

Run with `test_two_points.txt` and 2 processes with `valgrind`. Expect 0. Only penalize for new memory leaks.

**Test 8 (10 points)**

Run with `test_not_triangle.txt` and 2 processes. Expect 0; take off all points otherwise.

**Test 9 (15 points)**

Run with `test_giant_triangle.txt` and 2 processes. Expect 1; take off all points otherwise.

- 2 unused include/import
- 10 for poor commenting
- 2 for each unjustified numeric constant
- 5 bad submission
- 1 for each section of commented code
- 5 if violate coding convention