

CSE3221 2012-12S Assignment 1

Some notes on output/files:

- Where appropriate files/output are shown between dashed lines, e.g.

output/file goes here

- Where the correct answer is given, the correct standard output
is meant to be exact (but the number of decimal places doesn't matter).
Only a single number should be output. Anything else is extraneous.
The "correct" standard error is just an indication of an appropriate error message.
- Here are the files (f1 - f7) used in testing.
Files f1 - f4 are well-formed input files.
Files f5 - f7 are unexpected in that they either contain no tokens or contain some non-numeric data.

f1 (5 numbers. mean is 5.0)

1.1 2.2 3.3 4.4 14.0

f2 (5 numbers. mean is 10.0)

6.1 7.2 8.3 9.4 19.0

f3 (50,000 numbers. mean is 5.0)
This has 10,000 lines, each like the line in f1.

1.1 2.2 3.3 4.4 14.0
... 9,999 identical lines ...

f4 (50,000 numbers. mean is 10.0)
This has 10,000 lines, each like the line in f2.

6.1 7.2 8.3 9.4 19.0
... 9,999 identical lines ...

f5 (empty file: no bytes)

f6 (whitespace - blanks and tabs - only)

f7 (contains some non-numeric characters)

1.1 2.2 3.3 4.4NonNumeric 14.0

Testing process.c

Compiling with "cc -Wall -o prs process.c"
produced following message(s) to standard error (any such messages should have been dealt with):

process.c: In function `main':
process.c:22: warning: implicit declaration of function `exit'
process.c:73: warning: suggest parentheses around assignment used as truth value
process.c:110: warning: implicit declaration of function `wait'
process.c:28: warning: unused variable `fsp'

Testing "prs f1":

Correct program standard error: none

Tested program standard error: none

Correct program standard output:

5.000000

Tested program standard output:

5.000000

Testing "prs f1 f2":

Correct program standard error: none

Tested program standard error: none

Correct program standard output:

7.500000

Tested program standard output:

7.500000

Testing "prs f1 f1 f1 f1 f1 f2 f2 f2 f2 f2":

Correct program standard error: none

Tested program standard error: none

Correct program standard output:

7.500000

Tested program standard output:

7.500000

Testing "prs f3 f3 f3 f3 f3 f4 f4 f4 f4 f4":

Correct program standard error: none

Tested program standard error: none

Correct program standard output:

7.500000

Tested program standard output:

7.500000

Testing process.c unusual input/args

The following tests are with unexpected input.
Where there are no files on the command line, there should be a usage message written to standard error, not standard output.
No number/mean should be output.

Where there are no tokens read (f5 and f6) the result should be either

- silent return (no output)
- or
- a message to standard error (not standard output).

There should be no number output. The mean of no numbers is not 0 any more than it is -1 or 99.

Where there is non-numeric tokens in the file (f7), there should be no output to standard output. An error message should be written to standard error.

Testing "prs ":

Correct program standard error:

Usage: prs <file1> <file2> ...

Tested program standard error:

usage: prs <filename1> <filename2> ...

Correct program standard output: none

Tested program standard output: none

Testing "prs f5":

Correct program standard error: none

Tested program standard error: none

Correct program standard output: none

Tested program standard output:

0.000000

Testing "prs f6":

Correct program standard error: none

Tested program standard error: none

Correct program standard output: none

Tested program standard output:

0.000000

Testing "prs f7":

Correct program standard error:

Child 1. Error reading file "f7"
Some child had non-zero exit. Parent aborting.

Tested program standard error:

error in file: f7

Correct program standard output: none

Tested program standard output:

2.750000

Testing thread.c

Compiling with "cc -Wall -o thr thread.c -lpthread"
produced following message(s) to standard error (any such messages should have been dealt with):

thread.c: In function `main':
thread.c:30: warning: implicit declaration of function `exit'
thread.c:36: warning: implicit declaration of function `malloc'
thread.c:72: warning: implicit declaration of function `free'
thread.c: In function `calc_mean':
thread.c:84: warning: suggest parentheses around assignment used as truth value

Testing "thr f1":

Correct program standard error: none

Tested program standard error: none

Correct program standard output:

5.000000

Tested program standard output:

5.000000

Testing "thr f1 f2":

Correct program standard error: none

Tested program standard error: none

Correct program standard output:

7.500000

Tested program standard output:

7.500000

Testing "thr f1 f1 f1 f1 f1 f2 f2 f2 f2 f2":

Correct program standard error: none

Tested program standard error: none

Correct program standard output:

7.500000

Tested program standard output:

7.500000

Testing "thr f3 f3 f3 f3 f3 f4 f4 f4 f4 f4":

Correct program standard error: none

Tested program standard error: none

Correct program standard output:

7.500000

Tested program standard output:

7.500000

Testing thread.c unusual input/args

The following tests are with unexpected input.
Where there are no files on the command line, there should be a usage message written to standard error, not standard output.
No number/mean should be output.

Where there are no tokens read (f5 and f6) the result should be either

- silent return (no output)
- or
- a message to standard error (not standard output).

There should be no number output. The mean of no numbers is not 0 any more than it is -1 or 99.

Where there is non-numeric tokens in the file (f7), there should be no output to standard output. An error message should be written to standard error.

Testing "thr ":

Correct program standard error:

Usage: thr <file1> <file2> ...

Tested program standard error:

usage: thr <filename1> <filename2> ...

Correct program standard output: none

Tested program standard output: none

Testing "thr f5":

Correct program standard error: none

Tested program standard error: none

Correct program standard output: none

Tested program standard output:

0.000000

Testing "thr f6":

Correct program standard error: none

Tested program standard error: none

Correct program standard output: none

Tested program standard output:

0.000000

Testing "thr f7":

Correct program standard error:

Child 1. Error reading file "f7"
Some child had non-zero exit. Parent aborting.

Tested program standard error:

error in file: f7

Correct program standard output: none

Tested program standard output:

2.750000

COMMENTS
compiler warnings
incorrect output f5, f6, f7
process - sequential execution