Test Plan - Intersect

James Viner

Sample Files

`american-english.txt`

A copy of the american-english dictionary file.

`empty.txt`

An empty file

`headphonetic.txt`

The first ten lines from phonetic.txt with the last letter capitalized.

`phonetic.txt`

A-Z in the phonetic alphabet with the first letter capitalized, an empty line, then A-Z in the phonetic alphabet in all-uppercase.

`sameline.txt`

The first four words in the phonetic alphabet on the same line, with Charlie and Delta having their last letter capitalized.

`tailphonetic.txt`

The last 10 lines from phonetic.txt.

underscore.txt

"_Alpha", followed by an empty line, "Bravo_", followed by an empty line, "Charlie".

words.txt

A copy of the words dictionary file.

Automated Test Cases

Automated tests may be executed with `make check`, which builds and runs a bash script that performs unit tests for this program. Additional test details can be found in `test/test.bash`; the script may have to be made executable before running.

Common Test Cases

Prerequisites: `intersect/` directory does not already exist.

TC1: Installation

1. `git clone` git@git.umbc.tc:tdqc/tdqc12/jviner/intersect.git

2. `cd intersect`

Expected: `intersect/` directory is created.

TC2: Correct branch

1. git branch

Expected: main is default branch.

TC3: Build cleaning

1. `make`

2. `make clean`

3. `ls`

Expected: No .o files or executable files are left in directory.

Project-Building Test Cases

Prerequisites: In the project's directory; have run `make clean`.

TC4: Build Explicit

1. `make intersect`
2. `ls intersect`

Expected: File is listed.

TC5: Build Implicit

1. `make`
2. `ls intersect`

Expected: File is listed.

TC6: Build debugging

1. `make debug`
2. `readelf –debug-dump=decodedline intersect`

Expected: Debugging info is listed.

TC7: Build profiling

1. `make profile`

Expected: Program build output contains -gp flags.

File-Based Test Cases

Prerequisites: In the project's directory; have run `make`; sample files in `samp/` folder.

TC8: Leak Prevention - Very large files

1.
   ```
   valgrind ./intersect samp/american-english.txt samp/words.txt > /dev/null
   ```

Expected: Program does not enter infinite loop, exits normally, and valgrind shows no leaks.

TC9: Leak Prevention - Empty files

1.
   ```
   valgrind ./intersect samp/empty.txt samp/empty.txt
   ```

Expected: valgrind shows no leaks. Nothing is printed.

TC10: Leak Prevention - Nonexistent file

1.
   ```
   valgrind ./intersect samp/empty.txt samp/nonexistentfile.txt
   ```

Expected: valgrind shows no leaks. Appropriate error message printed.