

CS2303_HW05

This assignment tests creating dynamic-sized stacks and reading input from the keyboard. In total 4 programs were created:

1. `./stacktest` This takes x num strings, pushes them into a stack, and then pops them off.
2. `./stacktest_dbg` Same as `./stacktest` with debugging tools.
3. `./stacktest2` This takes x num strings, pushes them into a stack, and then prints each string in reverse order.
4. `./stacktest2_dbg` Same as `./stacktest2` with debugging tools.

Compilation

To compile the program, open the directory into any unix environment and enter make to compile all the source files.

```
$ cd <path to directory>/cs2303_hw05
~/<path to directory>/cs2303_hw05$ make
```

Note: To compile individual testers enter `make stacktest` or `make stacktest2`.

To get doxygen output, configure the target folder in the makefile and enter make docs.

```
$ make docs
```

To remove object files and executables, run this command:

```
$ make clean
```

stacktest usage

For normal output run:

```
$ ./stacktest max_elements
```

For enhanced output run:

```
$ ./stacktest_dbg max_elements
```

Parameter	Type	Description
<code>max_elements</code>	int	Required. Max number of elements to be stored in the stack.

stacktest2 usage

For normal output run:

```
$ ./stacktest2 max_elements
```

For enhanced output run:

```
$ ./stacktest2_dbg max_elements
```

Parameter	Type	Description
<code>max_elements</code>	int	Required. Max number of elements to be stored in the stack.

tests

```
$ ./stacktest
Enter exactly 1 argument into the command line!
```

```
$ ./stacktest -1
Entered -1 but max_elements must be >= 0!
```

```
$ ./stacktest 5 < input.txt
Fifth and final line
fourth
third line
This is the second line
This is a line
```

```
$ ./stacktest 3 < input.txt
third line
This is the second line
This is a line
```

```
$ ./stacktest 10 < input.txt
Fifth and final line
fourth
third line
This is the second line
This is a line
```

Note: Although only 5 elements were printed, this stack can contain 5 more elements.

```
$ ./stacktest2 5 < input.txt
enil lanif dna htfiF
htruof
enil driht
enil dnoces eht si sihT
enil a si sihT
```

Note: Only reverses non-null and non-newline characters.

```
$ cat input.txt | ./stacktest 100 | ./stacktest2 7
enil a si sihT
enil dnoces eht si sihT
enil driht
htruof
enil lanif dna htfiF
```

```
$ cat input.txt | ./stacktest2 100 | ./stacktest 7
enil a si sihT
enil dnoces eht si sihT
enil driht
htruof
enil lanif dna htfiF
```

Note: Exact same output as previous test.