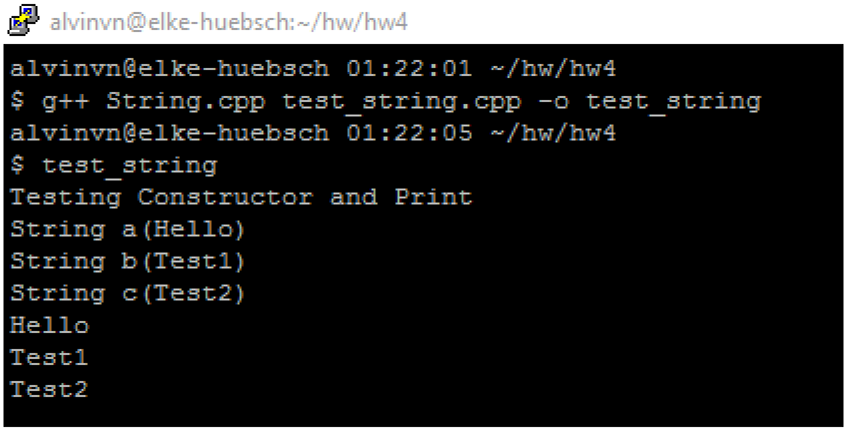
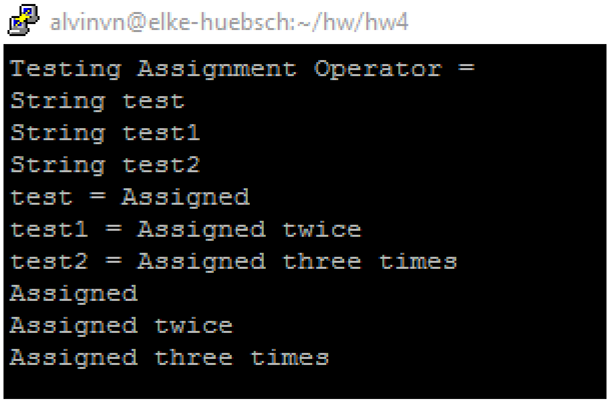
TEST_STRING:

Constructor:



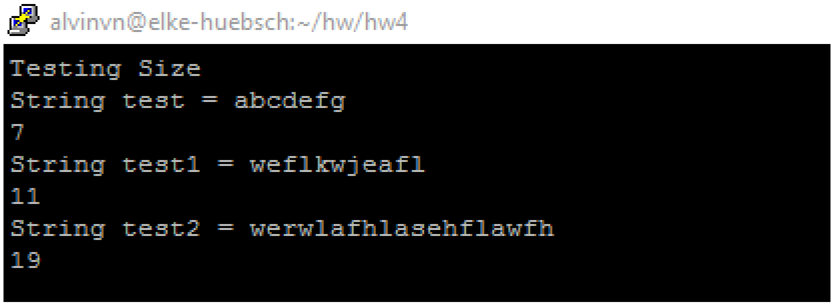
```
alvinvn@elke-huebsch 01:22:01 ~/hw/hw4
$ g++ String.cpp test_string.cpp -o test_string
alvinvn@elke-huebsch 01:22:05 ~/hw/hw4
$ test_string
Testing Constructor and Print
String a(Hello)
String b(Test1)
String c(Test2)
Hello
Test1
Test2
```

Assignment:



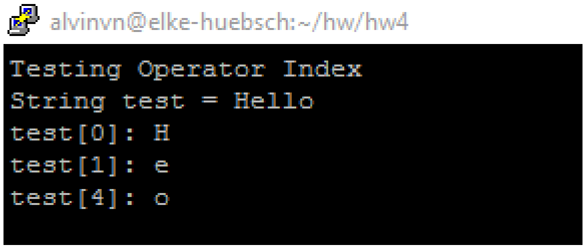
```
Testing Assignment Operator =
String test
String test1
String test2
test = Assigned
test1 = Assigned twice
test2 = Assigned three times
Assigned
Assigned twice
Assigned three times
```

Size:



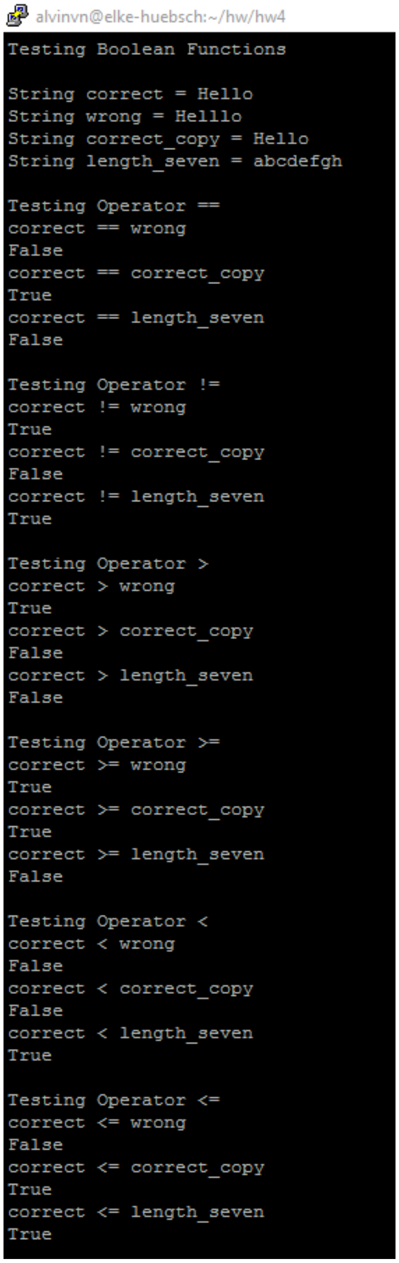
```
Testing Size
String test = abcdefg
7
String test1 = weflkwjeafl
11
String test2 = werwlafhlasehflawfh
19
```

Index Operator:



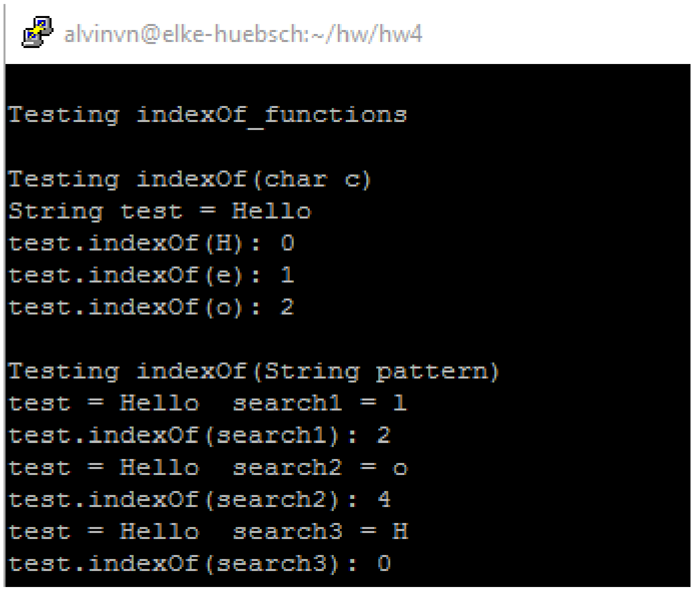
```
Testing Operator Index
String test = Hello
test[0]: H
test[1]: e
test[4]: o
```

Operators (==, !=, < , <= , > , =>):



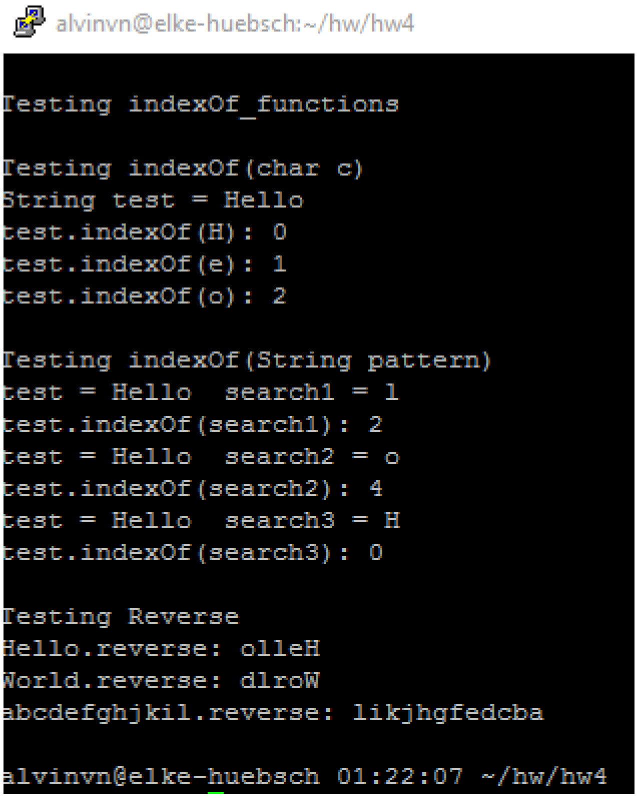
```
Testing Boolean Functions

String correct = Hello
String wrong = Helllo
String correct_copy = Hello
String length_seven = abcdefgh

Testing Operator ==
correct == wrong
False
correct == correct_copy
True
correct == length_seven
False

Testing Operator !=
correct != wrong
True
correct != correct_copy
False
correct != length_seven
True

Testing Operator >
correct > wrong
True
correct > correct_copy
False
correct > length_seven
False

Testing Operator >=
correct >= wrong
True
correct >= correct_copy
True
correct >= length_seven
False

Testing Operator <
correct < wrong
False
correct < correct_copy
False
correct < length_seven
True

Testing Operator <=
correct <= wrong
False
correct <= correct_copy
True
correct <= length_seven
True
```

IndexOf Functions:



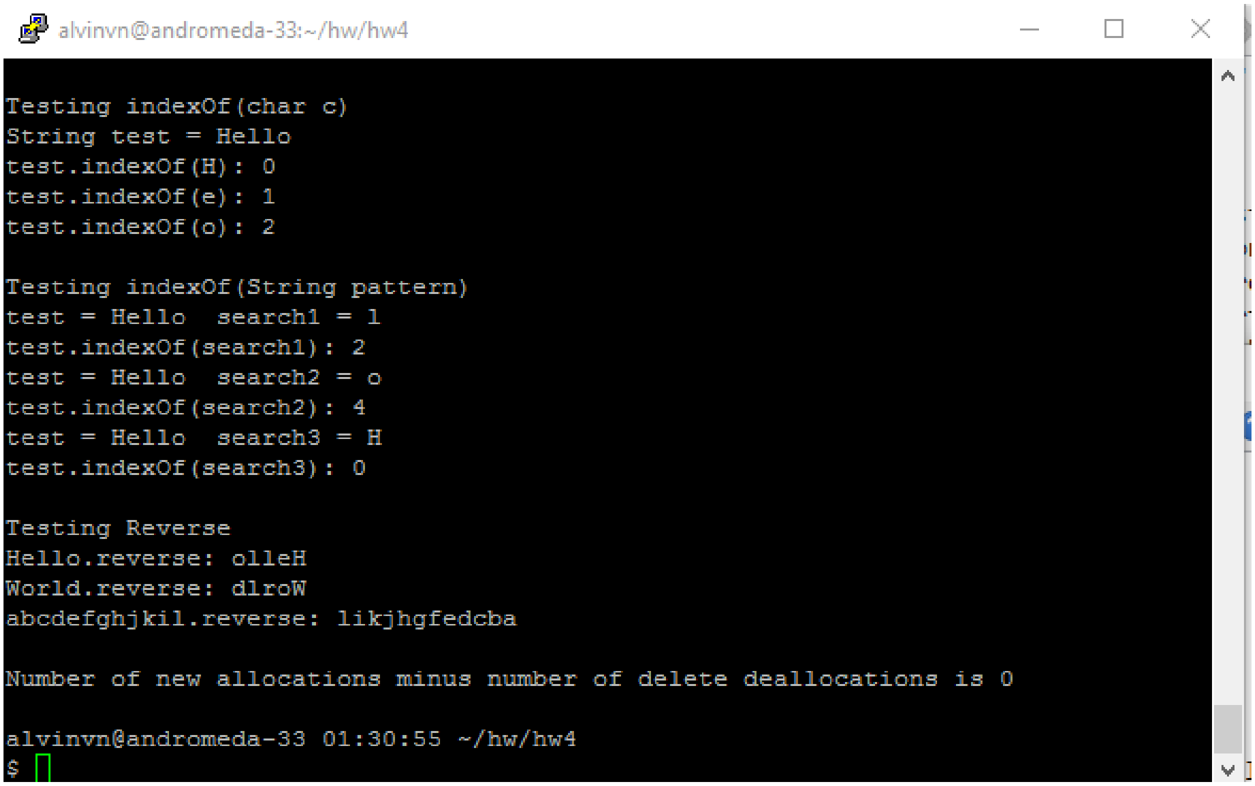
```
Testing indexOf_functions

Testing indexOf(char c)
String test = Hello
test.indexOf(H): 0
test.indexOf(e): 1
test.indexOf(o): 2

Testing indexOf(String pattern)
test = Hello  search1 = l
test.indexOf(search1): 2
test = Hello  search2 = o
test.indexOf(search2): 4
test = Hello  search3 = H
test.indexOf(search3): 0
```

Reverse:



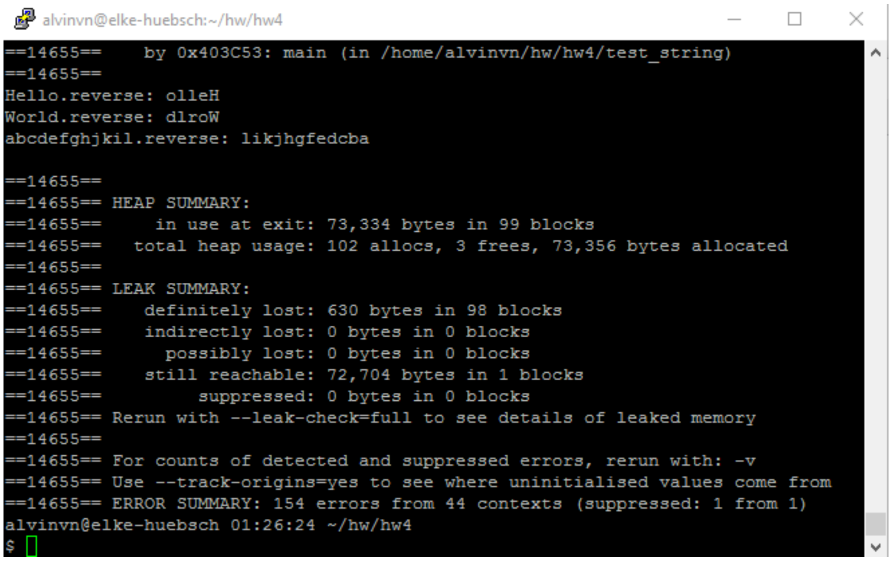
```
Testing indexOf_functions

Testing indexOf(char c)
String test = Hello
test.indexOf(H): 0
test.indexOf(e): 1
test.indexOf(o): 2

Testing indexOf(String pattern)
test = Hello  search1 = l
test.indexOf(search1): 2
test = Hello  search2 = o
test.indexOf(search2): 4
test = Hello  search3 = H
test.indexOf(search3): 0

Testing Reverse
Hello.reverse: olleH
World.reverse: dlroW
abcdefghjkil.reverse: likjhgfedcba

alvinvn@elke-huebsch 01:22:07 ~/hw/hw4
```

Allocations:



```
Testing indexOf(char c)
String test = Hello
test.indexOf(H): 0
test.indexOf(e): 1
test.indexOf(o): 2

Testing indexOf(String pattern)
test = Hello  search1 = l
test.indexOf(search1): 2
test = Hello  search2 = o
test.indexOf(search2): 4
test = Hello  search3 = H
test.indexOf(search3): 0

Testing Reverse
Hello.reverse: olleH
World.reverse: dlroW
abcdefghjkil.reverse: likjhgfedcba

Number of new allocations minus number of delete deallocations is 0

alvinvn@andromeda-33 01:30:55 ~/hw/hw4
$ 
```

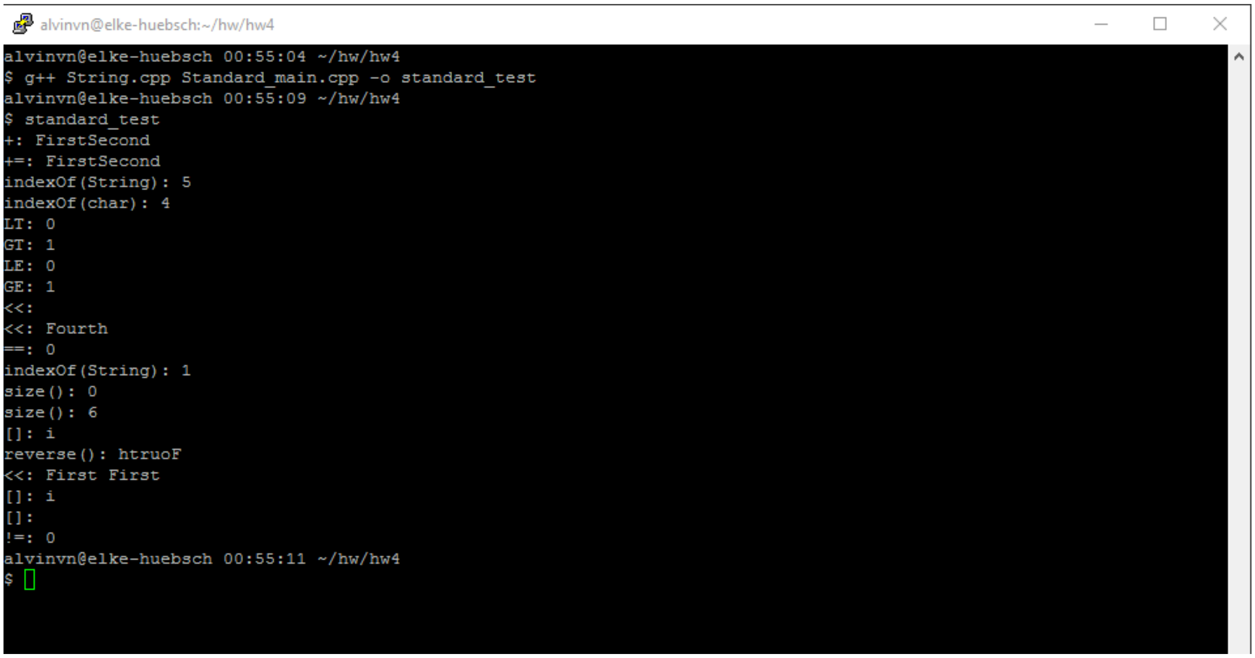
Valgrind Report:



```
==14655==    by 0x403C53: main (in /home/alvinvn/hw/hw4/test_string)
==14655==
Hello.reverse: olleH
World.reverse: dlroW
abcdefghjkil.reverse: likjhgfedcba

==14655==
==14655== HEAP SUMMARY:
==14655==     in use at exit: 73,334 bytes in 99 blocks
==14655==   total heap usage: 102 allocs, 3 frees, 73,356 bytes allocated
==14655==
==14655== LEAK SUMMARY:
==14655==    definitely lost: 630 bytes in 98 blocks
==14655==    indirectly lost: 0 bytes in 0 blocks
==14655==      possibly lost: 0 bytes in 0 blocks
==14655==    still reachable: 72,704 bytes in 1 blocks
==14655==         suppressed: 0 bytes in 0 blocks
==14655== Rerun with --leak-check=full to see details of leaked memory
==14655==
==14655== For counts of detected and suppressed errors, rerun with: -v
==14655== Use --track-origins=yes to see where uninitialised values come from
==14655== ERROR SUMMARY: 154 errors from 44 contexts (suppressed: 1 from 1)
alvinvn@elke-huebsch 01:26:24 ~/hw/hw4
$ 
```

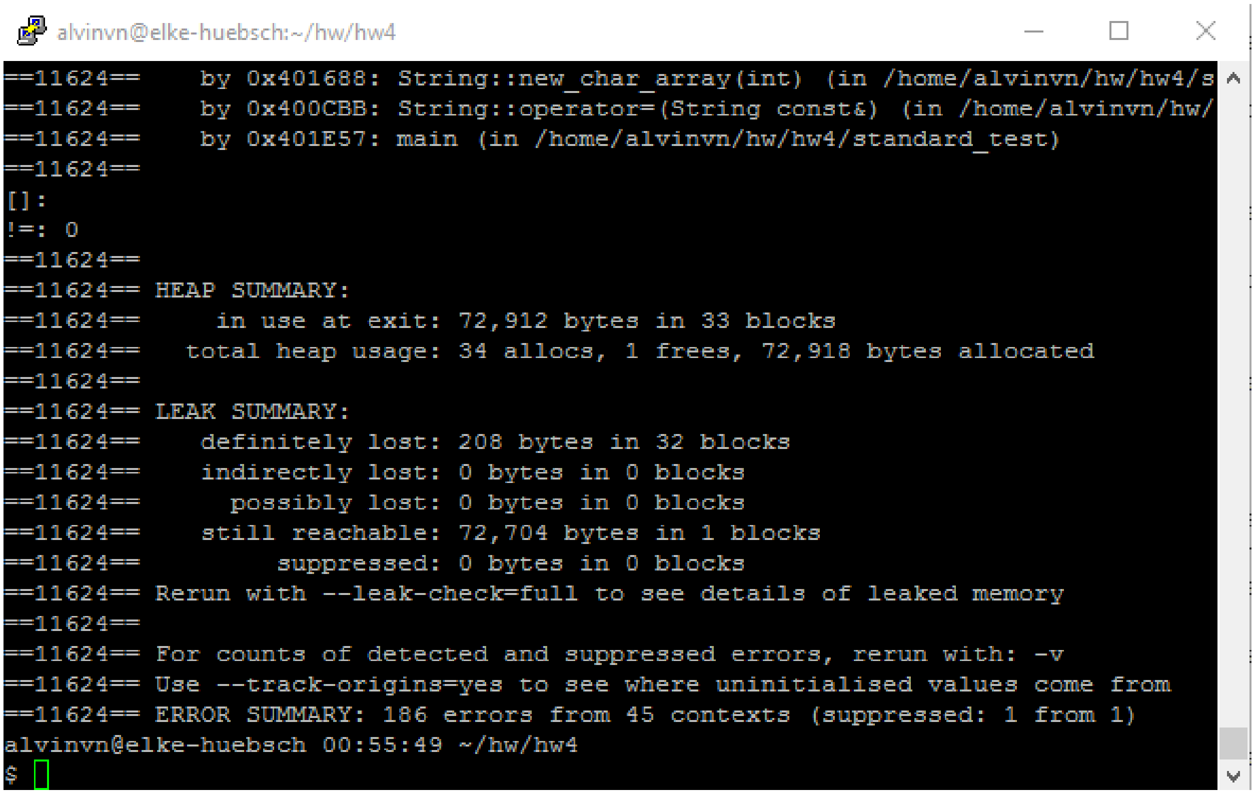
STANDARD_MAIN:

Test:



```
alvinvn@elke-huebsch 00:55:04 ~/hw/hw4
$ g++ String.cpp Standard_main.cpp -o standard_test
alvinvn@elke-huebsch 00:55:09 ~/hw/hw4
$ standard_test
+: FirstSecond
+=: FirstSecond
indexOf(String): 5
indexOf(char): 4
LT: 0
GT: 1
LE: 0
GE: 1
<<:
<<: Fourth
==: 0
indexOf(String): 1
size(): 0
size(): 6
[]: i
reverse(): htruoF
<<: First First
[]: i
[]:
!=: 0
alvinvn@elke-huebsch 00:55:11 ~/hw/hw4
$ 
```

Valgrind:



```
==11624==    by 0x401688: String::new_char_array(int) (in /home/alvinvn/hw/hw4/s
==11624==    by 0x400CBB: String::operator=(String const&) (in /home/alvinvn/hw/
==11624==    by 0x401E57: main (in /home/alvinvn/hw/hw4/standard_test)
==11624==
[]:
!=: 0
==11624==
==11624== HEAP SUMMARY:
==11624==     in use at exit: 72,912 bytes in 33 blocks
==11624==   total heap usage: 34 allocs, 1 frees, 72,918 bytes allocated
==11624==
==11624== LEAK SUMMARY:
==11624==    definitely lost: 208 bytes in 32 blocks
==11624==    indirectly lost: 0 bytes in 0 blocks
==11624==      possibly lost: 0 bytes in 0 blocks
==11624==    still reachable: 72,704 bytes in 1 blocks
==11624==         suppressed: 0 bytes in 0 blocks
==11624== Rerun with --leak-check=full to see details of leaked memory
==11624==
==11624== For counts of detected and suppressed errors, rerun with: -v
==11624== Use --track-origins=yes to see where uninitialised values come from
==11624== ERROR SUMMARY: 186 errors from 45 contexts (suppressed: 1 from 1)
alvinvn@elke-huebsch 00:55:49 ~/hw/hw4
$ 
```