Feedback

Points

Test	Points
test_FCFS_simple	0.1/0.1
test_HRRN_complex	0.5/0.5
test_HRRN_simple	0.5/0.5
test_LCFS_complex	0.25/0.25
test_LCFS_simple	0.25/0.25
test_MLF_complex	0.5/0.5
test_MLF_long	0.5/0.5
test_MLF_short	0.5/0.5
test_MLF_simple	0.5/0.5
test_PRIOP_complex	0/0.5
test_PRIOP_simple	0.5/0.5
test_RR_complex	0.5/0.5
test_RR_long	0.5/0.5
test_RR_short	0.5/0.5
test_RR_simple	0.5/0.5
test_SRTN_complex	0.5/0.5
test_SRTN_simple	0.5/0.5
test_queue_add	0.1/0.1
test_queue_complex	0.5/0.5
test_queue_free	0.1/0.1
test_queue_many	0.5/0.5
test_queue_new	0.1/0.1
test_queue_peek	0.1/0.1
test_queue_poll	0.1/0.1
test_queue_simple	0.4/0.4
memory leaks	0.4/0.5
deduction	-1.0
Total	8.4

Logs

deduction

reason:

```
submitted wrong files: -1
```

test_FCFS_simple.log

stdout:

```
Process count: 2
Strategy     : FCFS
+--------------------------------------+
| ID | Start Time | Duration | Priority |
+--------------------------------------+
|  A |          0 |        3 |        1 |
|  B |          2 |        2 |        3 |
+--------------------------------------+
Starting FCFS scheduler
 A | A | A | B | B |
Result
------
Got     : AAABB
Expected: AAABB
Matches
```

test_HRRN_complex.log

stdout:

```
Process count: 6
Strategy     : HRRN
+--------------------------------------+
| ID | Start Time | Duration | Priority |
+--------------------------------------+
|  A |          0 |        3 |        1 |
|  B |          1 |        8 |        3 |
|  C |          2 |        1 |        2 |
|  D |          3 |       10 |        1 |
|  E |          4 |        2 |        8 |
|  F |          5 |        1 |       22 |
+--------------------------------------+
Starting HRRN scheduler
 A | A | A | C | B | B | B | B | B | B | B | B | F | E | E | D | D | D | D | D | D | D | D | D | D |
Result
------
Got     : AAACBBBBBBBBFEEDDDDDDDDDD
Expected: AAACBBBBBBBBFEEDDDDDDDDDD
```

```
Matches
```

test_HRRN_simple.log

stdout:

```
Process count: 3
Strategy     : HRRN
+--------------------------------------+
| ID | Start Time | Duration | Priority |
+--------------------------------------+
|  A |          0 |        3 |        1 |
|  B |          1 |        2 |        4 |
|  C |          2 |        2 |        2 |
+--------------------------------------+
Starting HRRN scheduler
 A | A | A | B | B | C | C |
Result
------
Got     : AAABBCC
Expected: AAABBCC
Matches
```

test_LCFS_complex.log

stdout:

```
Process count: 7
Strategy     : LCFS
+--------------------------------------+
| ID | Start Time | Duration | Priority |
+--------------------------------------+
|  A |          0 |       10 |        1 |
|  B |          3 |        2 |        1 |
|  C |          4 |        3 |        1 |
|  D |          5 |        3 |        1 |
|  E |          6 |        8 |        1 |
|  F |         12 |        2 |        1 |
|  G |         20 |        2 |        1 |
+--------------------------------------+
Starting LCFS scheduler
 A | A | A | A | A | A | A | A | A | A | A | E | E | E | E | E | E | E | E | F | F | G | G | D | D | D | C | C | C | B | B |
Result
------
Got     : AAAAAAAAAAEEEEEEEEFFGGDDDCCCBB
Expected: AAAAAAAAAAEEEEEEEEFFGGDDDCCCBB
Matches
```

test_LCFS_simple.log

stdout:

```
Process count: 3
Strategy     : LCFS
+--------------------------------------+
| ID | Start Time | Duration | Priority |
+--------------------------------------+
|  A |          0 |        3 |        1 |
|  B |          1 |        2 |        4 |
|  C |          2 |        2 |        2 |
+--------------------------------------+
Starting LCFS scheduler
 A | A | A | C | C | B | B |
Result
------
Got     : AAACCBB
Expected: AAACCBB
Matches
```

test_MLF_complex.log

stdout:

```
Process count: 6
Strategy     : MLF
+--------------------------------------+
| ID | Start Time | Duration | Priority |
+--------------------------------------+
|  A |          0 |        3 |        4 |
|  B |          1 |        2 |        5 |
|  C |          2 |        5 |        6 |
|  D |          3 |        2 |        7 |
|  E |          6 |        4 |        8 |
|  F |          7 |        2 |        9 |
+--------------------------------------+
Starting MLF scheduler
 A | B | C | D | D | C | C | F | E | F | E | E | B | A | A | E | C | C |
Result
------
Got     : ABCDDCCFEFEEBAAECC
Expected: ABCDDCCFEFEEBAAECC
Matches
```

test_MLF_long.log

stdout:

```
Process count: 6
Strategy     : MLF
+--------------------------------------+
| ID | Start Time | Duration | Priority |
+--------------------------------------+
|  A |          0 |        3 |        4 |
|  B |          1 |        2 |        2 |
|  C |          4 |       28 |        6 |
|  D |          5 |        8 |        3 |
|  E |         18 |        3 |        8 |
|  F |         34 |        7 |        4 |
+--------------------------------------+
Starting MLF scheduler
 A | B | A | A | C | D | C | C | D | D | B | C | C | C | C | D | D | D | D | E | E | E | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | C | F | F | F | F | F | F | D |
Result
```

```
------
Got     : ABAACDCCDDBCCCCDDDDEEEECCCCCCCCCCCCCCCCCCCCCFFFFFFFD
Expected: ABAACDCCDDBCCCCDDDDEEEECCCCCCCCCCCCCCCCCCCCCFFFFFFFD
Matches
```

test_MLF_short.log

stdout:

```
Process count: 10
Strategy     : MLF
+---------------------------------------+
| ID | Start Time | Duration | Priority |
+---------------------------------------+
|  A |          0 |        1 |        4 |
|  B |          1 |        2 |        2 |
|  C |          2 |        3 |        2 |
|  D |          3 |        2 |        4 |
|  E |          4 |        4 |        2 |
|  F |          5 |        1 |        6 |
|  G |          6 |        2 |        1 |
|  H |          7 |        5 |        6 |
|  I |          8 |        2 |        3 |
|  J |          9 |        1 |        1 |
+---------------------------------------+
Starting MLF scheduler
 A | B | C | D | E | F | G | H | I | J | H | H | D | I | B | C | C | E | E | G | H | H | E |
Result
------
Got     : ABCDEFGHIJHHDIBCCEEGHHE
Expected: ABCDEFGHIJHHDIBCCEEGHHE
Matches
```

test_MLF_simple.log

stdout:

```
Process count: 3
Strategy     : MLF
+---------------------------------------+
| ID | Start Time | Duration | Priority |
+---------------------------------------+
|  A |          0 |        3 |        1 |
|  B |          1 |        2 |        4 |
|  C |          2 |        2 |        2 |
+---------------------------------------+
Starting MLF scheduler
 A | B | C | B | C | A | A |
Result
------
Got     : ABCBCAA
Expected: ABCBCAA
Matches
```

test_PRIOP_complex.log

stdout:

```
Process count: 7
Strategy     : PRIOP
+---------------------------------------+
| ID | Start Time | Duration | Priority |
+---------------------------------------+
|  A |          0 |        3 |        3 |
|  B |          1 |        1 |        5 |
|  C |          2 |        1 |        4 |
|  D |          3 |        8 |        1 |
|  E |          6 |        2 |        2 |
|  F |          7 |        2 |        3 |
|  G |         10 |        2 |        4 |
+---------------------------------------+
Starting PRIOP scheduler
 A | B | C | A | A | D | E | F | F | E | G | G | E | E | E | E | E | E | E |
Result
------
Got     : ABCAADEFFEGGEEEEEEE
Expected: ABCAADEFFEGGDDDDDDD
Did not match!
```

test_PRIOP_simple.log

stdout:

```
Process count: 3
Strategy     : PRIOP
+---------------------------------------+
| ID | Start Time | Duration | Priority |
+---------------------------------------+
|  A |          0 |        3 |        1 |
|  B |          1 |        2 |        4 |
|  C |          2 |        2 |        2 |
+---------------------------------------+
Starting PRIOP scheduler
 A | B | B | C | C | A | A |
Result
------
Got     : ABBCCAA
Expected: ABBCCAA
Matches
```

test_RR_complex.log

stdout:

```
Process count: 5
Strategy     : RR
Quantum      : 3
+---------------------------------------+
| ID | Start Time | Duration | Priority |
+---------------------------------------+
|  A |          0 |        3 |        1 |
|  B |          4 |        8 |        4 |
|  C |          9 |        7 |        4 |
```

```
|  D |        13 |        2 |        4 |
|  E |        15 |        4 |        2 |
+-----------------------------------------+
Starting RR scheduler
 A | A | A |  | B | B | B | B | B | B | C | C | C | B | B | D | D | C | C | C | E | E | E | C | E |
Result
------
Got     : AAA BBBBBBCCCBBDDCCCEEECE
Expected: AAA BBBBBBCCCBBDDCCCEEECE
Matches
```

test_RR_long.log

stdout:

```
Process count: 5
Strategy     : RR
Quantum      : 7
+-----------------------------------------+
| ID | Start Time | Duration | Priority |
+-----------------------------------------+
|  A |          0 |        7 |        1 |
|  B |          1 |       23 |        1 |
|  C |          2 |        3 |        1 |
|  D |          4 |       18 |        1 |
|  E |          6 |        4 |        1 |
+-----------------------------------------+
Starting RR scheduler
 A | A | A | A | A | A | A | A | B | B | B | B | B | B | B | C | C | C | D | D | D | D | D | D | D | E | E | E | E | B | B | B | B | B | B | B
| D | D | D | D | D | D | D | B | B | B | B | B | B | B | D | D | D | D | B | B |
Result
------
Got     : AAAAAAABBBBBBBCCCDDDDDDDEEEEBBBBBBBDDDDDDDBBBBBBBDDDDB
Expected: AAAAAAABBBBBBBCCCDDDDDDDEEEEBBBBBBBDDDDDDDBBBBBBBDDDDB
Matches
```

test_RR_short.log

stdout:

```
Process count: 8
Strategy     : RR
Quantum      : 1
+-----------------------------------------+
| ID | Start Time | Duration | Priority |
+-----------------------------------------+
|  A |          0 |        3 |        1 |
|  B |          1 |        1 |        1 |
|  C |          2 |        1 |        1 |
|  D |          6 |        4 |        1 |
|  E |          7 |        2 |        1 |
|  F |          9 |        3 |        1 |
|  G |         20 |        5 |        1 |
|  H |         22 |        3 |        1 |
+-----------------------------------------+
Starting RR scheduler
 A | B | A | C | A |  | D | E | D | E | F | D | F | D | F |  |  |  |  | G | G | H | G | H | G | H | G |
Result
------
Got     : ABACA DEDEFDFDF     GGHGHGHG
Expected: ABACA DEDEFDFDF     GGHGHGHG
Matches
```

test_RR_simple.log

stdout:

```
Process count: 3
Strategy     : RR
Quantum      : 2
+-----------------------------------------+
| ID | Start Time | Duration | Priority |
+-----------------------------------------+
|  A |          0 |        3 |        1 |
|  B |          1 |        2 |        4 |
|  C |          2 |        2 |        2 |
+-----------------------------------------+
Starting RR scheduler
 A | A | B | B | C | C | A |
Result
------
Got     : AABBCCA
Expected: AABBCCA
Matches
```

test_SRTN_complex.log

stdout:

```
Process count: 6
Strategy     : SRTN
+-----------------------------------------+
| ID | Start Time | Duration | Priority |
+-----------------------------------------+
|  A |          0 |        5 |        1 |
|  B |          1 |        2 |        1 |
|  C |          2 |        3 |        1 |
|  D |          5 |        1 |        1 |
|  E |          7 |        4 |        1 |
|  F |          8 |        2 |        1 |
+-----------------------------------------+
Starting SRTN scheduler
 A | B | B | C | C | C | D | A | F | F | A | A | A | E | E | E | E |
Result
------
Got     : ABBCCCDAFFAAAEEEE
Expected: ABBCCCDAFFAAAEEEE
Matches
```

test_SRTN_simple.log

stdout:

```
Process count: 3
Strategy     : SRTN
+--------------------------------------+
| ID | Start Time | Duration | Priority |
+--------------------------------------+
|  A |          0 |        3 |        1 |
|  B |          1 |        2 |        4 |
|  C |          2 |        2 |        2 |
+--------------------------------------+
Starting SRTN scheduler
 A | A | A | B | B | C | C |
Result
------
Got     : AAABBCC
Expected: AAABBCC
Matches
```

test_queue_add.log

stdout:

```
Creating a single queue with new_queue
Adding a single element with queue_add
Checking if next of queue head is not NULL
Checking if next->next is NULL
Test passed
```

test_queue_complex.log

stdout:

```
There are 26 letters in the alphabet. Let's shuffle them (using the alphabet in reversed order)
Creating three queues
Moving first half of alphabet into queue_one
Moving second half of alphabet into queue_two
Merging from queue_one and queue_two into queue_three
Polling from queue_three and creating new string
Freeing all three queues
Result: zmylxkwjviuhtgsfreqdpcobna
Comparing new string to expected output
Test passed
```

test_queue_free.log

stdout:

```
Creating a single queue with new_queue
Freeing queue again
Test passed
```

test_queue_many.log

stdout:

```
Creating 10 queues
Adding 100 elements to each queue with special values in between
Polling first 49 elements from each queue
Peeking all queues, expecting special value
Polling all queues, expecting special value
Polling another 48 elements from each queue
Polling all queues, expecting special value
Freeing all queues (with one element still left over)
Test passed
```

stderr:

```

Memcheck, a memory error detector
Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
Command: test_queue_many


HEAP SUMMARY:
    in use at exit: 160 bytes in 10 blocks
  total heap usage: 1,011 allocs, 1,001 frees, 20,256 bytes allocated

160 bytes in 10 blocks are definitely lost in loss record 1 of 1
   at 0x483B7F3: malloc (in x86_64-linux-vgpreload_memcheck-amd64-linux.so)
   by 0x10A971: new_queue (in test_queue_many)
   by 0x1091C7: main (in test_queue_many)

LEAK SUMMARY:
   definitely lost: 160 bytes in 10 blocks
   indirectly lost: 0 bytes in 0 blocks
     possibly lost: 0 bytes in 0 blocks
   still reachable: 0 bytes in 0 blocks
        suppressed: 0 bytes in 0 blocks

For lists of detected and suppressed errors, rerun with: -s
ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
```

test_queue_new.log

stdout:

```
Creating a single queue with new_queue
Checking next of new queue head is NULL
Test passed
```

test_queue_peek.log

stdout:

```
Creating a single queue with new_queue
Peeking queue and checking value
Adding a single element with queue_add
Peeking queue and checking value
Peeking queue again
Test passed
```

test_queue_poll.log

stdout:

```
Creating a single queue with new_queue
Adding a single element with queue_add
```

```
Polling queue and checking value
Polling queue again
Test passed
```

test_queue_simple.log

stdout:

```
Creating new queue
Adding A to queue
Peeking queue
Adding B to queue
Polling queue
Polling queue
Peeking queue (should be empty)
Polling queue (should be empty)
Freeing queue
Test passed
```

memory leaks.log

stderr:

```

Leak in test_queue_many, -0.1 Points
```