

9. Q4

```
$ cat looping-race-nolock.s
# assumes %bx has loop count in it

.main
.top
# critical section
mov 2000, %ax # get 'value' at address 2000
add $1, %ax   # increment it
mov %ax, 2000 # store it back

# see if we're still looping
sub $1, %bx
test $0, %bx
jgt .top

halt

$ python ./x86.py -p looping-race-nolock.s -t 1 -a bx=1 -M 2000 -c

2000      Thread 0
0
0  1000 mov 2000, %ax
0  1001 add $1, %ax
1  1002 mov %ax, 2000
1  1003 sub $1, %bx
1  1004 test $0, %bx
1  1005 jgt .top
1  1006 halt

$ python ./x86.py -p looping-race-nolock.s -t 2 -a bx=1 -M 2000 -R bx -c

2000      bx      Thread 0      Thread 1
0          1
0          1  1000 mov 2000, %ax
0          1  1001 add $1, %ax
1          1  1002 mov %ax, 2000
1          0  1003 sub $1, %bx
1          0  1004 test $0, %bx
1          0  1005 jgt .top
1          0  1006 halt
1          1  ----- Halt;Switch ----- ----- Halt;Switch -----
1          1          1000 mov 2000, %ax
1          1          1001 add $1, %ax
2          1          1002 mov %ax, 2000
2          0          1003 sub $1, %bx
2          0          1004 test $0, %bx
2          0          1005 jgt .top
2          0          1006 halt
```