

#####

Script started on Tue, Nov 11, 2014 2:13:53 PM

justin@Justin ~/cs333/p6

```
$ make && blitz -g os
kpl Kernel -unsafe
asm Kernel.s
lddd Runtime.o Switch.o System.o List.o BitMap.o Kernel.o Main.o -o os
kpl TestProgram3 -unsafe
asm TestProgram3.s
lddd UserRuntime.o UserSystem.o TestProgram3.o -o TestProgram3
diskUtil -i
diskUtil -a MyProgram MyProgram
diskUtil -a TestProgram1 TestProgram1
diskUtil -a TestProgram2 TestProgram2
diskUtil -a TestProgram3 TestProgram3
Beginning execution...
```

===== KPL PROGRAM STARTING =====

Initializing Thread Scheduler...

Initializing Thread Manager...

Initializing Frame Manager...

AllocateRandomFrames called. NUMBER_OF_PHYSICAL_PAGE_FRAMES = 512

Initializing Disk Driver...

Initializing File Manager...

SysExitTest running.

About to terminate the only process; should cause the OS to stop on a 'wait' instruction.

***** A 'wait' instruction was executed and no more interrupts are scheduled... halting emulation! *****

Done! The next instruction to execute will be:

```
001138: 09000000      ret
Number of Disk Reads   = 11
Number of Disk Writes  = 0
Instructions Executed   = 3192082
Time Spent Sleeping    = 9729
Total Elapsed Time     = 3201811
```

justin@Justin ~/cs333/p6

\$ exit

Script done on Tue, Nov 11, 2014 2:14:05 PM

Script started on Tue, Nov 11, 2014 2:14:17 PM

```

justin@Justin ~/cs333/p6

$ make && blitz -g os
kpl TestProgram3 -unsafe
asm TestProgram3.s
lddd UserRuntime.o UserSystem.o TestProgram3.o -o TestProgram3
diskUtil -i
diskUtil -a MyProgram MyProgram
diskUtil -a TestProgram1 TestProgram1
diskUtil -a TestProgram2 TestProgram2
diskUtil -a TestProgram3 TestProgram3
Beginning execution...
===== KPL PROGRAM STARTING =====

Initializing Thread Scheduler...

Initializing Thread Manager...

Initializing Frame Manager...

AllocateRandomFrames called.  NUMBER_OF_PHYSICAL_PAGE_FRAMES = 512

Initializing Disk Driver...

Initializing File Manager...


BasicForkTest running.


I am the parent

I am the child


**** A 'wait' instruction was executed and no more interrupts are scheduled... halting
emulation! ****


Done! The next instruction to execute will be:
001138: 09000000      ret
Number of Disk Reads   = 11
Number of Disk Writes  = 0
Instructions Executed   = 3456712
Time Spent Sleeping    = 9729
    Total Elapsed Time = 3466441


justin@Justin ~/cs333/p6

$ exit
Script done on Tue, Nov 11, 2014  2:14:25 PM

Script started on Tue, Nov 11, 2014  2:14:34 PM


justin@Justin ~/cs333/p6

$ make && blitz -g os
kpl TestProgram3 -unsafe
asm TestProgram3.s
lddd UserRuntime.o UserSystem.o TestProgram3.o -o TestProgram3
diskUtil -i
diskUtil -a MyProgram MyProgram
diskUtil -a TestProgram1 TestProgram1
diskUtil -a TestProgram2 TestProgram2
diskUtil -a TestProgram3 TestProgram3

```

```
Beginning execution...
===== KPL PROGRAM STARTING =====

Initializing Thread Scheduler...

Initializing Thread Manager...

Initializing Frame Manager...

AllocateRandomFrames called.  NUMBER_OF_PHYSICAL_PAGE_FRAMES = 512

Initializing Disk Driver...

Initializing File Manager...


YieldTest running.


This test involves calls to Fork, Yield, and Exit.


RUN ONE: You should see 10 'compiler' messages and 10 'OS' messages.


Designing compilers is fun!
Designing compilers is fun!
Designing compilers is fun!
Designing compilers is fun!
Designing compilers is fun!
Designing compilers is fun!
Designing compilers is fun!
Designing compilers is fun!
Designing compilers is fun!
Designing compilers is fun!

Writing OS kernel code is a blast!
Writing OS kernel code is a blast!
Writing OS kernel code is a blast!
Writing OS kernel code is a blast!
Writing OS kernel code is a blast!
Writing OS kernel code is a blast!
Writing OS kernel code is a blast!
Writing OS kernel code is a blast!
Writing OS kernel code is a blast!
Writing OS kernel code is a blast!
```

RUN TWO: You should see the same 20 messages, but the order should be different, due to the presence of 'Yield's.

```
Designing compilers is fun!
Designing compilers is fun!
Writing OS kernel code is a blast!
Designing compilers is fun!
Designing compilers is fun!
Writing OS kernel code is a blast!
Designing compilers is fun!
Writing OS kernel code is a blast!
Designing compilers is fun!
Writing OS kernel code is a blast!
Writing OS kernel code is a blast!
Designing compilers is fun!
Writing OS kernel code is a blast!
Writing OS kernel code is a blast!
Writing OS kernel code is a blast!
Designing compilers is fun!
Writing OS kernel code is a blast!
Writing OS kernel code is a blast!
Designing compilers is fun!
Designing compilers is fun!
```

```
***** A 'wait' instruction was executed and no more interrupts are scheduled... halting
emulation! *****
```

```
Done! The next instruction to execute will be:
001138: 09000000      ret
Number of Disk Reads   = 11
Number of Disk Writes  = 0
Instructions Executed   = 4006981
Time Spent Sleeping    = 9729
    Total Elapsed Time = 4016710
```

```
justin@Justin ~/cs333/p6
```

```
$ exit
```

```
Script done on Tue, Nov 11, 2014  2:14:45 PM
```

```
Script started on Tue, Nov 11, 2014  2:14:57 PM
```

```
justin@Justin ~/cs333/p6
```

```
$ make && blitz -g os
kpl TestProgram3 -unsafe
asm TestProgram3.s
lddd UserRuntime.o UserSystem.o TestProgram3.o -o TestProgram3
diskUtil -i
diskUtil -a MyProgram MyProgram
diskUtil -a TestProgram1 TestProgram1
diskUtil -a TestProgram2 TestProgram2
diskUtil -a TestProgram3 TestProgram3
Beginning execution...
===== KPL PROGRAM STARTING =====

Initializing Thread Scheduler...

Initializing Thread Manager...

Initializing Frame Manager...

AllocateRandomFrames called.  NUMBER_OF_PHYSICAL_PAGE_FRAMES = 512

Initializing Disk Driver...

Initializing File Manager...
```

ForkTest running.

This test involves calls to Fork, Yield, and Exit.

There should be 26 columns (A-Z) printed. Each letter should be printed 5 times.

A

A

A

B

B

A

B

B

A

C

C

B

C

C

D

D

D

C

D

xx

SKIPPED

xx

W

W

V

W

W

X

X

W

X

X

Y

Y

X

Y

Y

Z

Z

Y

Z

Z

Z

***** A 'wait' instruction was executed and no more interrupts are scheduled... halting emulation! *****

Done! The next instruction to execute will be:

```
001138: 09000000      ret
Number of Disk Reads   = 11
Number of Disk Writes  = 0
Instructions Executed   = 51961662
Time Spent Sleeping    = 9729
      Total Elapsed Time = 51971391
justin@Justin ~/cs333/p6
```

\$ exit

Script done on Tue, Nov 11, 2014 2:15:07 PM

Script started on Tue, Nov 11, 2014 2:15:29 PM

justin@Justin ~/cs333/p6

\$ make && blitz -g os

```

kpl TestProgram3 -unsafe
asm TestProgram3.s
lddd UserRuntime.o UserSystem.o TestProgram3.o -o TestProgram3
diskUtil -i
diskUtil -a MyProgram MyProgram
diskUtil -a TestProgram1 TestProgram1
diskUtil -a TestProgram2 TestProgram2
diskUtil -a TestProgram3 TestProgram3
Beginning execution...
===== KPL PROGRAM STARTING =====

Initializing Thread Scheduler...

Initializing Thread Manager...

Initializing Frame Manager...

AllocateRandomFrames called.  NUMBER_OF_PHYSICAL_PAGE_FRAMES = 512

Initializing Disk Driver...

Initializing File Manager...


JoinTest 1 running.


This test involves calls to Fork, Yield, and Exit.

Running first test...

This line should print first.

This line should print second.

Done.

Running second test...

This line should print first.

This line should print second.

Done.


***** A 'wait' instruction was executed and no more interrupts are scheduled... halting
emulation! *****


Done! The next instruction to execute will be:
001138: 09000000      ret
Number of Disk Reads   = 11
Number of Disk Writes  = 0
Instructions Executed   = 4342505
Time Spent Sleeping    = 9729
Total Elapsed Time     = 4352234


justin@Justin ~/cs333/p6

$ exit
Script done on Tue, Nov 11, 2014  2:15:36 PM

Script started on Tue, Nov 11, 2014  2:15:46 PM

```

justin@Justin ~/cs333/p6

```
$ make && blitz -g os
kpl TestProgram3 -unsafe
asm TestProgram3.s
lddd UserRuntime.o UserSystem.o TestProgram3.o -o TestProgram3
diskUtil -i
diskUtil -a MyProgram MyProgram
diskUtil -a TestProgram1 TestProgram1
diskUtil -a TestProgram2 TestProgram2
diskUtil -a TestProgram3 TestProgram3
Beginning execution...
===== KPL PROGRAM STARTING =====

Initializing Thread Scheduler...

Initializing Thread Manager...

Initializing Frame Manager...

AllocateRandomFrames called.  NUMBER_OF_PHYSICAL_PAGE_FRAMES = 512

Initializing Disk Driver...

Initializing File Manager...


JoinTest 2 running.


This test involves calls to Fork, Yield, and Exit.

Creating 5 children...

Child 1 running...

Child 2 running...

Child 3 running...

Child 4 running...

Waiting for children in order 1, 2, 3, 4, 5...

Child 5 running...

Creating 5 more children...

Child 1 running...

Child 2 running...

Child 3 running...

Child 4 running...

Waiting for children in order 5, 4, 1, 3, 2...

Child 5 running...

Done.


***** A 'wait' instruction was executed and no more interrupts are scheduled... halting
emulation! *****
```



```
Done! The next instruction to execute will be:
001138: 09000000      ret
Number of Disk Reads   = 11
Number of Disk Writes  = 0
Instructions Executed   = 5859892
Time Spent Sleeping    = 9729
Total Elapsed Time     = 5869621
```

```
justin@Justin ~/cs333/p6
```

```
$ exit
```

```
Script done on Tue, Nov 11, 2014  2:16:11 PM
```

```
Script started on Tue, Nov 11, 2014  2:21:41 PM
```

```
justin@Justin ~/cs333/p6
```

```
$ make && blitz -g os
kpl TestProgram3 -unsafe
asm TestProgram3.s
lddd UserRuntime.o UserSystem.o TestProgram3.o -o TestProgram3
diskUtil -i
diskUtil -a MyProgram MyProgram
diskUtil -a TestProgram1 TestProgram1
diskUtil -a TestProgram2 TestProgram2
diskUtil -a TestProgram3 TestProgram3
Beginning execution...
===== KPL PROGRAM STARTING =====

Initializing Thread Scheduler...

Initializing Thread Manager...

Initializing Frame Manager...

AllocateRandomFrames called.  NUMBER_OF_PHYSICAL_PAGE_FRAMES = 512

Initializing Disk Driver...

Initializing File Manager...

JoinTest3 running.
```

This test involves 5 illegal calls to Sys_Join, waiting on non-existent children.

In each case, it prints the return code, which should be -1.

Return code from 1st call = -1

Return code from 2nd call = -1

Return code from 3rd call = -1

Return code from 4th call = -1

Return code from 5th call = -1

Done.

```
***** A 'wait' instruction was executed and no more interrupts are scheduled... halting
emulation! *****
```

```
Done! The next instruction to execute will be:
001138: 09000000      ret
Number of Disk Reads   = 11
Number of Disk Writes  = 0
Instructions Executed   = 3199905
Time Spent Sleeping    = 9729
Total Elapsed Time     = 3209634
```

```
justin@Justin ~/cs333/p6
```

```
$ exit
```

```
Script done on Tue, Nov 11, 2014  2:21:57 PM
```

```
Script started on Tue, Nov 11, 2014  2:22:22 PM
```

```
justin@Justin ~/cs333/p6
```

```
$ make && blitz -g os
kpl TestProgram3 -unsafe
asm TestProgram3.s
lddd UserRuntime.o UserSystem.o TestProgram3.o -o TestProgram3
diskUtil -i
diskUtil -a MyProgram MyProgram
diskUtil -a TestProgram1 TestProgram1
diskUtil -a TestProgram2 TestProgram2
diskUtil -a TestProgram3 TestProgram3
Beginning execution...
===== KPL PROGRAM STARTING =====

Initializing Thread Scheduler...

Initializing Thread Manager...

Initializing Frame Manager...

AllocateRandomFrames called.  NUMBER_OF_PHYSICAL_PAGE_FRAMES = 512

Initializing Disk Driver...

Initializing File Manager...


JoinTest4 running.
```

This test forks a child process and then waits on it twice.

The first call to Sys_Join should return its error code; the
second call to Sys_Join should return -1.

The PID of the child = 2

This should print first.

This should print second.

Okay (1).

Okay (2).

This should print first.
The PID of the child = 3
This should print second.
Okay (3).
Okay (4).

In the next test, we create 2 children, and each creates 2 children,
giving 7 processes in all. Then each process attempts a Sys_Join on
every process except its own children, to make sure the result is -1.
Finally, each process with children waits on them.

A is running...
My first child is A.B pid1 = 4
My second child is A.C pid2 = 5

A.B.D is running...

A.C.F is running...

A.C is running...
My first child is A.C.F pid1 = 7
My second child is A.C.G pid2 = 9

A.C.G is running...

A.B is running...
My first child is A.B.D pid1 = 6
My second child is A.B.E pid2 = 8

A.B.E is running...

A done with error tests...
A.C.F done with error tests...
A.C.G done with error tests...
A.C done with error tests...

```

A.B done with error tests...
A.B.D done with error tests...
A.B.E done with error tests...

-----A is waiting on A.B      pid1 = 4
-----A.C is waiting on A.C.F    pid1 = 7
-----A.B is waiting on A.B.D    pid1 = 6
A.C.F is done.
-----A.C is waiting on A.C.G    pid2 = 9
A.C.G is done.
A.C is done.
A.B.E is done.
A.B.D is done.
-----A.B is waiting on A.B.E    pid2 = 8
A.B is done.
-----A is waiting on A.C      pid2 = 5
A is done.

***** A 'wait' instruction was executed and no more interrupts are scheduled... halting
emulation! *****

Done! The next instruction to execute will be:
001138: 09000000      ret
Number of Disk Reads   = 11
Number of Disk Writes  = 0
Instructions Executed   = 18123758
Time Spent Sleeping    = 9729
Total Elapsed Time     = 18133487

justin@Justin ~/cs333/p6

$ exit
Script done on Tue, Nov 11, 2014  2:22:35 PM

Script started on Tue, Nov 11, 2014  2:22:50 PM

justin@Justin ~/cs333/p6

$ make && blitz -g os
kpl TestProgram3 -unsafe
asm TestProgram3.s
ldd UserRuntime.o UserSystem.o TestProgram3.o -o TestProgram3
diskUtil -i
diskUtil -a MyProgram MyProgram
diskUtil -a TestProgram1 TestProgram1
diskUtil -a TestProgram2 TestProgram2
diskUtil -a TestProgram3 TestProgram3
Beginning execution...
===== KPL PROGRAM STARTING =====

Initializing Thread Scheduler...

```

Initializing Thread Manager...

Initializing Frame Manager...

AllocateRandomFrames called. NUMBER_OF_PHYSICAL_PAGE_FRAMES = 512

Initializing Disk Driver...

Initializing File Manager...

ManyProcessesTest1 running.

This test should create 100 child processes.

It should print 100 lines of output.

Child 1

Child 2

Child 3

Child 4

Child 5

Child 6

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

SKIPPED

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

Child 89

Child 90

Child 91

Child 92

Child 93

Child 94

Child 95

Child 96

Child 97

Child 98

Child 99

Child 100

Done.

***** A 'wait' instruction was executed and no more interrupts are scheduled... halting emulation! *****

Done! The next instruction to execute will be:

001138: 09000000 ret

Number of Disk Reads = 11

Number of Disk Writes = 0

```
Instructions Executed   = 29498796
Time Spent Sleeping    = 9729
Total Elapsed Time     = 29508525
```

```
justin@Justin ~/cs333/p6
```

```
$ exit
```

```
Script done on Tue, Nov 11, 2014  2:23:00 PM
```

```
Script started on Tue, Nov 11, 2014  2:23:11 PM
```

```
justin@Justin ~/cs333/p6
```

```
$ make && blitz -g os
kpl TestProgram3 -unsafe
asm TestProgram3.s
lddd UserRuntime.o UserSystem.o TestProgram3.o -o TestProgram3
diskUtil -i
diskUtil -a MyProgram MyProgram
diskUtil -a TestProgram1 TestProgram1
diskUtil -a TestProgram2 TestProgram2
diskUtil -a TestProgram3 TestProgram3
Beginning execution...
===== KPL PROGRAM STARTING =====
```

```
Initializing Thread Scheduler...
```

```
Initializing Thread Manager...
```

```
Initializing Frame Manager...
```

```
AllocateRandomFrames called.  NUMBER_OF_PHYSICAL_PAGE_FRAMES = 512
```

```
Initializing Disk Driver...
```

```
Initializing File Manager...
```

```
ManyProcessesTest2 running.
```

```
This test attempts to create 9 new processes.
```

```
It should print a line for each process and then it should print 123.
```

```
Process 0
```

```
Process 1
```

```
Process 2
```

```
Process 3
```

```
Process 4
```

```
Process 5
```

```
Process 6
```

```
Process 7
```

```
Process 8
```

```
Process 9
```

```
Final return value = 123
```

Done.

***** A 'wait' instruction was executed and no more interrupts are scheduled... halting emulation! *****

Done! The next instruction to execute will be:

```
001138: 09000000      ret
Number of Disk Reads   = 11
Number of Disk Writes  = 0
Instructions Executed   = 5728919
Time Spent Sleeping    = 9729
Total Elapsed Time     = 5738648
```

justin@Justin ~/cs333/p6

\$ exit

Script done on Tue, Nov 11, 2014 2:23:21 PM

Script started on Tue, Nov 11, 2014 2:23:29 PM

justin@Justin ~/cs333/p6

\$ make && blitz -g os

kpl TestProgram3 -unsafe

asm TestProgram3.s

lddd UserRuntime.o UserSystem.o TestProgram3.o -o TestProgram3

diskUtil -i

diskUtil -a MyProgram MyProgram

diskUtil -a TestProgram1 TestProgram1

diskUtil -a TestProgram2 TestProgram2

diskUtil -a TestProgram3 TestProgram3

Beginning execution...

===== KPL PROGRAM STARTING =====

Initializing Thread Scheduler...

Initializing Thread Manager...

Initializing Frame Manager...

AllocateRandomFrames called. NUMBER_OF_PHYSICAL_PAGE_FRAMES = 512

Initializing Disk Driver...

Initializing File Manager...

ManyProcessesTest3 running.

This test attempts to create 10 new processes.

It should run out of resources and hang.

Process 0

Process 1

Process 2

Process 3

Process 4
Process 5
Process 6
Process 7
Process 8
Process 9

***** A 'wait' instruction was executed and no more interrupts are scheduled... halting emulation! *****

Done! The next instruction to execute will be:
001138: 09000000 ret
Number of Disk Reads = 11
Number of Disk Writes = 0
Instructions Executed = 5478764
Time Spent Sleeping = 9729
Total Elapsed Time = 5488493

justin@Justin ~/cs333/p6

\$ exit

Script done on Tue, Nov 11, 2014 2:23:38 PM

Script started on Tue, Nov 11, 2014 2:23:47 PM

justin@Justin ~/cs333/p6

```
$ make && blitz -g os
kpl TestProgram3 -unsafe
asm TestProgram3.s
lddd UserRuntime.o UserSystem.o TestProgram3.o -o TestProgram3
diskUtil -i
diskUtil -a MyProgram MyProgram
diskUtil -a TestProgram1 TestProgram1
diskUtil -a TestProgram2 TestProgram2
diskUtil -a TestProgram3 TestProgram3
Beginning execution...
===== KPL PROGRAM STARTING =====

Initializing Thread Scheduler...

Initializing Thread Manager...

Initializing Frame Manager...

AllocateRandomFrames called.  NUMBER_OF_PHYSICAL_PAGE_FRAMES = 512

Initializing Disk Driver...

Initializing File Manager...

ErrorTest running.
```

Should print "User Program Error: Attempt to use a null pointer!"...

User Program Error: Attempt to use a null pointer! Type 'st' to see stack.

Okay.

Should print "An AddressException exception has occurred while in user mode"...

***** An AddressException exception has occurred while in user mode *****

```
ProcessControlBlock (addr=0x00029520) pid=3, status=ACTIVE, parentsPid=1, exitStatus=0
```

	addr	entry	Logical	Physical	Undefined Bits	Dirty	Referenced
	Writeable	Valid					
	=====	=====	=====	=====	=====	=====	=====
	=====	=====					
YES	0x0002954C:	0x00154005	0x00000000	0x00154000			YES
YES	0x00029550:	0x00158001	0x00002000	0x00158000			
YES	0x00029554:	0x0015C001	0x00004000	0x0015C000			
YES	0x00029558:	0x00160001	0x00006000	0x00160000			
YES	0x0002955C:	0x00164005	0x00008000	0x00164000			YES
YES	0x00029560:	0x00168003	0x0000A000	0x00168000			
YES	0x00029564:	0x0016C003	0x0000C000	0x0016C000			
YES	0x00029568:	0x00170003	0x0000E000	0x00170000			
YES	0x0002956C:	0x00174003	0x00010000	0x00174000			
YES	0x00029570:	0x0017800F	0x00012000	0x00178000		YES	YES
myThread = Thread "UserProgramThread" status=RUNNING (addr of Thread object: 0x0002BB68)							

Thread "UserProgramThread" (addr of Thread object: 0x0002BB68)

machine state:

r2: 0x00000000 0
r3: 0x00000000 0
r4: 0x00000000 0

```
r5: 0x00000000  0
r6: 0x00000000  0
r7: 0x00000000  0
r8: 0x00000000  0
r9: 0x00000000  0
r10: 0x00000000  0
r11: 0x00000000  0
r12: 0x00000000  0
r13: 0x00000000  0
r14: 0x00000000  0

stackTop = 0x0002CB5C
stack starting addr = 0x0002BBC4
status = RUNNING
is user thread: TRUE
user registers:
r1: 0x00000000  0
r2: 0x00000000  0
r3: 0x00000000  0
r4: 0x00000000  0
r5: 0x00000004  4
r6: 0x00000000  0
r7: 0x00000000  0
r8: 0x00000000  0
r9: 0x00000000  0
r10: 0x00004345 17221
r11: 0x00000000  0
r12: 0x00000000  0
r13: 0x0000003A  58
r14: 0x00013FA0 81824
r15: 0x00013F7C 81788
```

Okay.

Should print "A PageReadonlyException exception has occurred while in user mode"...

***** A PageReadOnlyException exception has occurred while in user mode *****

```

ProcessControlBlock (addr=0x000295C8) pid=4, status=ACTIVE, parentsPid=1, exitStatus=0

      addr      entry      Logical      Physical      Undefined Bits      Dirty      Referenced
Writeable Valid
=====
0x000295F4: 0x00154005 0x00000000 0x00154000
YES
0x000295F8: 0x00158001 0x00002000 0x00158000
YES
0x000295FC: 0x0015C001 0x00004000 0x0015C000
YES
0x00029600: 0x00160001 0x00006000 0x00160000
YES
0x00029604: 0x00164005 0x00008000 0x00164000
YES
0x00029608: 0x00168003 0x0000A000 0x00168000
YES YES
0x0002960C: 0x0016C003 0x0000C000 0x0016C000
YES YES
0x00029610: 0x00170003 0x0000E000 0x00170000
YES YES
0x00029614: 0x00174003 0x00010000 0x00174000
YES YES
0x00029618: 0x0017800F 0x00012000 0x00178000
YES YES YES
myThread = Thread "UserProgramThread" status=RUNNING (addr of Thread object:
0x0002CBAC)

```

Thread "UserProgramThread" (addr of Thread object: 0x0002CBAC)

machine state:

```

r2: 0x000283C0 164800
r3: 0x00000004 4
r4: 0x00000000 0
r5: 0x00000000 0
r6: 0x00000000 0
r7: 0x00000000 0
r8: 0x00000000 0
r9: 0x00000000 0
r10: 0x00004345 17221
r11: 0x00000000 0
r12: 0x00000000 0
r13: 0x00000052 82
r14: 0x0002DACC 187084

```

stackTop = 0x0002DA70

stack starting addr = 0x0002CC08

status = RUNNING

is user thread: TRUE

user registers:

r1:	0x00000000	0
r2:	0x00000000	0
r3:	0x00000000	0
r4:	0x00000000	0
r5:	0x00000004	4
r6:	0x00000000	0
r7:	0x00000000	0
r8:	0x00000000	0
r9:	0x00000000	0
r10:	0x00004345	17221
r11:	0x00000000	0
r12:	0x00000000	0
r13:	0x0000003A	58
r14:	0x00013FA0	81824
r15:	0x00013F7C	81788

Okay.

Done.

***** A 'wait' instruction was executed and no more interrupts are scheduled... halting emulation! *****

Done! The next instruction to execute will be:

001138:	09000000	ret
Number of Disk Reads	=	11
Number of Disk Writes	=	0
Instructions Executed	=	4033401
Time Spent Sleeping	=	9729
Total Elapsed Time	=	4043130

justin@Justin ~/cs333/p6

\$ exit

Script done on Tue, Nov 11, 2014 2:23:58 PM