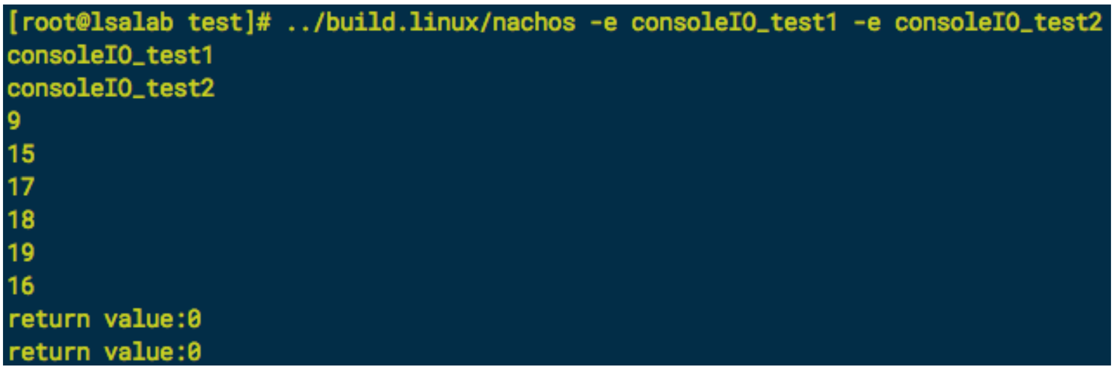
CS342301: Operating System
MP2: Multi-Programming
Deadline: 2017/11/20 08:00

I. Goal

1. Understand how memory management works in NachOS
2. Understand how to implement page table mechanism

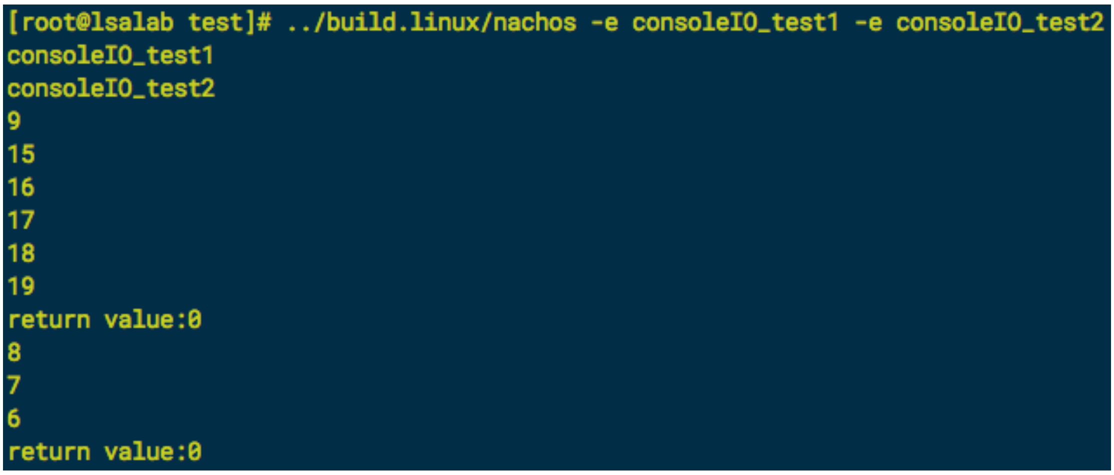
II. Assignment

1. Implement page table in NachOS
   - The NachOS doesn't support multi-programming now, you have to modify the code to make NachOS support it.
     - ➢ Wrong results without multi-programming



```
[root@lsalab test]# ../build.linux/nachos -e consoleIO_test1 -e consoleIO_test2
consoleIO_test1
consoleIO_test2
9
15
17
18
19
16
return value:0
return value:0
```

     - ➢ Correct results with multi-programming



```
[root@lsalab test]# ../build.linux/nachos -e consoleIO_test1 -e consoleIO_test2
consoleIO_test1
consoleIO_test2
9
15
16
17
18
19
return value:0
8
7
6
return value:0
```

III. Instruction

1. Copy MP1 to a new folder
   - cp –r NachOS-4.0_MP1 NachOS-4.0_MP2
2. Test your program
   - cd NachOS-4.0_MP2/code/test

- ../build.linux/nachos –e **consoleIO_test1** –e **consoleIO_test2**

3. Reminder
    - Comment out lines calling Halt() in **consoleIO_test1.c** and **consoleIO_test2.c**
    - Use “ctrl + c” to terminate

IV. Grading

1. Implementation correctness – 60%
    - Execute “../build.linux/nachos –e **consoleIO_test1** –e **consoleIO_test2**” correctly
2. Trace code – 25%
    - Goal: understand how NachOS creates a thread(process)
    - Include your writeup in the report file
    - Trace must be starting from “**Kernel::ExecAll()**”, until “**Machine::Run()**” is called for executing the first instruction from user program.
    - You should explain the purpose of each function in the call path, and make sure your explanation provides the answer for the following questions.
        - How Nachos allocates the memory space for new thread(process)?
        - How Nachos initializes the memory content of a thread(process), including loading the user binary code in the memory?
        - How Nachos creates and manages the page table?
        - How Nachos translates address?
        - How Nachos initializes the machine status (registers, etc) before running a thread(process)
        - Which Nachos **object** acts the role of **process control block**
        - When and how does a thread get added into the ReadyToRun queue of Nachos CPU scheduler?

3. Report – 15%
    - Explain how you modified the code & group contribution
    - Filename: MP2_[yourGroupNumber].pdf

V. Hint

- The following files “may” be modified…
    - userprog/addrspace.*
    - threads/kernel.*

VI. Reminder

1. iLMS
    (a). Upload your Report in **PDF** format to iLMS.
    (b). You **DO NOT** need to upload your NachOS code to iLMS, but we will use your latest modification time as your submission time.
2. Demo policy
    (a). Demo will take place on our server.
    (b). You are responsible to make sure your code works on our server.
    (c). Limit 10 mins for each team, so please be well prepared for it.
3. Refer to syllabus for late submission penalty.
4. 0 will be given to cheaters.