

EECS 211
Advanced System Software
Winter 2005

Assignment 4

Posted: March 7, 2005
Due: March 14, 2005

Topic, Option 1: Software-managed TLB in Nachos

Topic, Option 2: Operating System Development and Deployment

Instructions:

Choose either one of the two options. Detailed instructions for each option are outlined on the following pages.

--

Rainer Doemer (ET 444C, x4-9007, doemer@uci.edu)

Instructions for Option 1: Software-managed TLB in Nachos

The goal of this assignment is to develop, implement and test support for a software-managed TLB in the Nachos system. This assignment follows the “Nachos Assignment 3” described in the file `doc/vm.ps` of the Nachos installation. The instructions below assume that you read `doc/vm.ps` in parallel. Note that, because of the limited time frame, we will only do step 1 of this assignment, namely replace the hardware-supported page tables with a software-managed TLB. Thus, no virtual memory support is necessary (all pages stay in the memory of the machine at all times).

Option 1, Task 1: Understand the given framework

Go into the `vm` directory. Notice that there is no new source code provided. All code is given with the previous assignment. Simply copy the files that need to be modified from the `userprog` directory over into the `vm` directory.

Option 1, Task 2: Implement the software-management for the TLB

See item 1 in `doc/vm.ps`.

Modify your code from the previous assignment such that it compiles and runs fine when given the `-DUSE_TLB` flags. In particular, you will need to adjust the files `exception.cc` and `addrspace.cc` (and possibly others).

To test your changes, re-use your test cases `reverse.c`, `bullet.c` and the shell. You may also use the provided `matmult.c` and `sort.c` programs as test cases, but you will probably need to increase your main memory size for them to run. Provide appropriate log files as evidence that your implementation works.

Option 1, Deliverables:

1. ASCII text file `assignment4.txt` explaining your experiments; this should be no longer than one page!
2. Files `exception.cc`, `addrspace.cc`, and any other modified files.
3. Log files, i.e. `shell.log`

We will use the same electronic submission procedure as in the previous assignments. Please refer to the instructions listed with Assignment 3 for details.

Instructions for Option 2: OS development and deployment

The idea of this assignment is to put yourself into the shoes of Bill Gates (MS Windows), Linus Torvalds (Linux), or other leading operating system developers in the world. The central question is, what does it take to repeat a similar success story, given that you would need to compete with them.

Option 2, Task 1: OS development

List and describe the features necessary in your OS to make it successful. Distinguish between standard options that are simply necessary, and special features that would give your OS an advantage over others. Also, indicate how to structure your development so that your new OS can be deployed in reasonable time.

Option 2, Task 2: OS deployment

Discuss how you would deploy your operating system such that it attracts users and developers of application software. Note that a large user base is critical to the success of your new OS.

Option 2, Deliverables:

1. ASCII text file `assignment4.txt` outlining your new OS and its features, development and deployment strategy; this should be no longer than two pages!

We will use the same electronic submission procedure as in the previous assignments. To submit your text file, put it into a folder called `vm` on the server and use the usual `turnin` script. Please refer to the instructions listed with Assignment 3 for details.