CST 334: Operating Systems

Dr. Glenn Bruns

# OSTEP Chapter 16

**Purpose**. In segmented memory the virtual address space has multiple segments, but a base-and-bounds approach is used for each segment. The purpose of this assignment is to help you understand segmentation and to be able to perform address translation under this virtual memory approach.

**Instructions**. Read read OSTEP chapter 16, and answer the following questions by downloading and editing [**chap16.txt**](https://drive.google.com/file/d/1jfA-8GWGCmHdbazG870J8YLQmKFqdbge/view?usp=sharing).

1. Suppose we have a segmented virtual address space, with 2 segments. If a virtual address uses 1 bit for the segment, and 7 bits for the offset, how big is the virtual address space? (As in problem 1a, please give your answer as a integer without units.)

256

1. (T/F) Virtual addresses from two different processes must never translate to the same physical address.
2. (T/F) With segmentation, the number of segments used is never more than 3.
3. (T/F) As a programmer, it is helpful to know the order of the code, heap, and stack segments in physical memory.
4. (T/F) Segmentation helps in the problem of sharing code between processes.
5. (T/F) The address space of each segment must be the same size.

Suppose we are using segmentation as an extension of simple base-and-bounds address translation. Suppose, like in Chapter 16 of OSTEP, that we have segments “Code”, “Heap”, and “Stack”, where the base and bounds of each segment are as follows (all values are decimal):

|  |  |  |
| --- | --- | --- |
| Segment | Base | Bound |
| Code | 1000 | 4096 |
| Heap | 16000 | 2048 |
| Stack | 8000 | 2048 |

For each of the following virtual addresses, give the physical address (assume addresses “grow positive” in each segment), or that a segmentation fault has occurred. Provide the physical address as a decimal number.

1. (Code, 153)

1153

1. (Heap, 3327)

fault

1. (Code, 6211)

fault

1. (Stack, 32)

8032

**Submitting.** Submit your edited chap16.txt on iLearn.

**Grading**. Each problem is worth 10 points.