**CIS 350 – INFRASTRUCTURE TECHNOLOGIES**

**SMALL GROUP ACTIVITY #7 (EXTRA CREDIT)**

Names: Brooklynn Taylor, Dylan Lasley, Jonathan McCarrick, Zoe Druen

**Topic: Address translation under segmentation, paging, and segmentation and**

**Paging (Translation of virtual addresses to physical addresses)**

Logistics

1. Get in touch with your group. (See Groups folder on Blackboard.)
2. Discuss and complete the assignment together via E-mail, Discussion Forum, Blackboard Collaborate Ultra, and/or MS Teams.
3. Choose a recorder to prepare the final copy (one per group) and submit it via the Blackboard Assignments/Small Group Activities folder to the instructor.
4. Be sure all group members' names are on final copy. Do not add names of your group classmates who did not participate in the assignment.

I. Assignment

Work 3 problems on address translation under (1) segmentation, (2) paging, and (3) segmentation and paging.

In particular, your job has the following sections:

A. Allocate space for the program when it is loaded

1. Find the memory space to use

2. Create the appropriate segment or page tables

B. Perform address translation during I-time (Instruction time)

1. Calculate the physical address from the virtual address under

Segmentation, paging, and segmentation and paging

Note: Clarification on I-time. In the computer, instructions are

executed in machine cycles. The machine cycle consists of I-time (Instruction time – Fetch Phase) and E-time (Execution time – Execution Phase). During I-time the instruction is fetched from memory to the instruction register (Fetch Phase). During E-time the instruction is executed (Execution Phase). Address translation is performed during I-time before the instruction is executed. I-time works with virtual addresses, whereas E-time operates on physical addresses.

HANDY CHART TO HELP WITH ADDRESSING

0K = 0 7K = 7168 14K = 14336

1K = 1024 8K = 8192 15K = 15360

2K = 2048 9K = 9216 16K = 16384

3K = 3072 10K = 10240 17K = 17408

4K = 4096 11K = 11264 18K = 18432

5K = 5120 12K = 12288 19K = 19456

6K = 6144 13K = 13312 20K = 20480

MEMORY MAP FOR SEGMENTATION

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| | | |

| Start Address | Length | Status |

‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑

| 20K | 8K | 1 |

| 28K | 16K | ~~0~~ **1** | seg #0

| 44K | 16K | 1 |

| 60K | 4K | ~~0~~  **1** | seg #1

| 64K | 12K | ~~0~~ **1**  | seg #2

| 76K | 10K | 1  |

| 86K | 12K | ~~0~~ **1** | seg #3

| 98K | 10K | 1 |

|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|

PROBLEM 1

1. Load Program A ‑ 4 segments: seg 0 = size 15K, 1=4K, 2=6K, 3=10K

2. Create Segment Table

|  |  |
| --- | --- |
| Seg # | Starting address |
| 0 | 28672 |
| 1 | 61440 |
| 2 | 65536 |
| 3 | 88064 |

\_\_\_\_\_\_\_\_\_\_\_

3. Find the physical address of virtual address | 0 | 50 |

‑‑‑‑‑‑‑‑‑‑‑

seg# disp.

28672 + 50 = 68722

PAGE FRAME TABLE FOR PAGING

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| | | | |

| PAGE | PROGRAM ID | PAGE | STATUS |

| FRAME # | | NUMBER | |

|\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_|

| 0 | Operating Sys | 0 | 1 |

| 1 | Operating Sys | 1 | 1 |

| 2 | Operating Sys | 2 | 1 |

| 3 | Operating Sys | 3 | 1 |

|\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_|

| 4 | Program X | 0 | 1 |

| 5 | Program X | 1 | 1 |

| 6 | **program A** | **0** | ~~0~~ **1** |page 0

| 7 | Program Y | 0 | 1 |

|\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_|

| 8 | Program X | 2 | 1 |

| 9 | Program X | 3 | 1 |

| 10 | **program A**  | **1**  | ~~0~~ **1**  |page 1

| 11 | Program X | 4 | 1 |

|\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_|

| 12 | Program Y | 1 | 1 |

| 13 | **program A**  | **2**  | ~~0~~ **1** |page 2

| 14 | **program A** | **3**  | ~~0~~ **1** |page 3

| 15 | **program A** | **4**  | ~~0~~ **1** |page 4

|\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_|

NOTE: Each frame is 4K, so the address of Page Frame #4 would be

16K (4K \* Page Frame #)

PROBLEM 2

1. Load Program A ‑ 20K

2. Create Page Table

|  |  |
| --- | --- |
| Page # | Starting address |
| 0 | 24576 |
| 1 | 40960 |
| 2 | 53248 |
| 3 | 57344 |
| 4 | 61440 |

\_\_\_\_\_\_\_\_\_\_\_

3. Find the physical address of virtual address | 3 | 100 |

‑‑‑‑‑‑‑‑‑‑‑

page# disp

57344 + 100 = 57444

PAGE FRAME TABLE FOR SEGMENTATION & PAGING

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| | | | | |

| PAGE | PROGRAM ID |SEGMENT | PAGE | STATUS |

| FRAME # | | NUMBER | NUMBER | |

|\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_|

| 0 | Operating Sys | 0 | 0 | 1 |

| 1 | Operating Sys | 0 | 1 | 1 |

| 2 | Operating Sys | 1 | 0 | 1 |

| 3 | Operating Sys | 1 | 1 | 1 |

|\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_|

| 4 | Program X | 0 | 0 | 1 |

| 5 | Program Y | 0 | 0 | 1 |

| 6 | **Program A** | **0** | **0** | ~~0~~ 1 | seg#0, page 0

| 7 | **Program A**  | **0**  | **1**  | ~~0~~ 1 | seg#0, page 1

|\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_|

| 8 | Program X | 0 | 1 | 1 |

| 9 | Program X | 1 | 0 | 1 |

| 10 | Program X | 1 | 1 | 1 |

| 11 | Program A | **1**  | **0**  | ~~0~~ 1 | seg#1, page 0

|\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_|

| 12 | Program A | **2**  | **0**  | ~~0~~ 1 | seg#2, page 0

| 13 | Program A | **2** | **1** | ~~0~~ 1 | seg#2, page 1

| 14 | Program Y | 1 | 0 | 1 |

| 15 | Program A | **2**  | **2**  | ~~0~~ 1 | seg#2, page 2

|\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_|

| 16 | Program Y | 1 | 1 | 1 |

| 17 | Program X | 2 | 0 | 1 |

| 18 | Program A | **2** | **3** | ~~0~~ 1 | seg#2, page 3

| 19 | Program A | **3**  | **0**  | ~~0~~ 1 | seg#3, page 0

|\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_|

NOTE: Each frame is 4K, so the address of Page Frame #4 would be

16K (4K \* Page Frame #)

PROBLEM 3

1. Load Program A ‑ 4 segments: seg 0 = size 8K, 1=4K, 2=16K, 3=4K

2. Create Page Table for Seg #2 only

|  |  |
| --- | --- |
| page # | Starting address |
| 0 | 49152 |
| 1 | 53248 |
| 2 | 61440 |
| 3 | 73728 |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3. Find the physical address of virtual address | 2 | 3 | 110 |

-‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑‑-

seg# page# disp.

73728 + 110 = 73838