Assignment 4

This assignment is due on December 1.

1. Given the following segment table, answer the questions below. Assume the units are all bytes.

Segment	Base Address	Length	Actual Size
0	4000	1000	960
1	400	600	576
2	0	200	192
3	1200	250	224
4	1500	950	928
5	2500	1500	1472

   (a) (5%) Assuming total physical memory is 5120 bytes, what is the external fragmentation as a percentage of total memory?

   (b) (5%) What is the total internal fragmentation as a fraction of total allocated memory?

   (c) (10%) For each of the logical addresses below expressed in the form (*segment id, offset*), write the physical address to which it maps, or state whether there is an en error and what type.

      i. (0,956)
      ii. (0,160)
      iii. (3, 200)
      iv. (5, 1480)
      v. (2, 160)

2. (10%) Draw a diagram that shows how a virtual memory system with combined segmentation and paging translates logical to physical addresses, assuming that the machine also has a TLB for fast translation of linear addresses and a cache that stores data by logical address. Assume that single-level paging is used, that segment and page tables are kept in memory, and that the processor has a Segment Table Base Register (STBR). If you plan to submit electronically, do not waste time making it pretty; you can hand draw it, scan it, and insert the scanned image as a PDF.

3. (20%) A machine has a physical memory with 2^{32} addressable bytes and a page size of 8 KB. Each process is allocated a virtual address space of 4 GB. Page table entries are 32 bits long. Page tables are kept in pageable memory.

   (a) Why is one-level paging inadequate for this system?

   (b) Why is two-level paging sufficient?

   (c) How many bits are needed to reference the outer page table and how many to reference the inner page table? Explain your answer showing all appropriate arithmetic.

   (d) Draw a flow chart showing how a logical address is translated to a physical address using this system, assuming it does not have a TLB.

4. (30%) Given the following reference string,

   1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6

   how many page faults would occur for the following replacement algorithms and the given number of frames allocated to the process:

(a) LRU, with both 5 frames and then 6 frames

(b) FIFO, with 5 then 6 frames

(c) Optimal, with 5, then 6 frames.

5. (20%) Consider a 100 by 100 two-dimensional array A of 4-byte integers allocated dynamically, as with the pseudo code instruction

```
int** A = new int [100] [100]
```

Assume that arrays are stored in row-major order, meaning that all of array 0 is stored in consecutive bytes, then row 1, then row 2, and so on. Also assume that the array is stored starting in logical address 1000, and that page size is 200 bytes. Suppose the following small loop fragment is entirely located within logical addresses 0 to 199, so that it is in page 0 of the process. Therefore, each instruction is fetched from page 0.

```
for ( i = 0; i < 100; i++ )
    for ( j = 0; j < 100; j++ )
        A[i][j] = 0;
```

How many page faults will occur in the execution of this loop if the process is allocated three frames and page 0 is in frame 1, and LRU is used for page replacement?

Submitting the Solution

You must type your assignment. Handwritten assignments will not be accepted. You may, if you wish, submit it electronically instead of handing it in. This will save paper. If you choose to do this, then login to eniac remotely or go to Lab 1000G and login there and follow these instructions:

- The file must be either a plain text file or a PDF document;
- It must be named *hwk4_username* (with a .pdf extension if it is a PDF file);
- It must be placed in the directory `/data/yoda/b/student.accounts/cs340/projects/hwk4`;
- It must have permission 0600. If you (still) do not know how to do this, use the command `chmod 0600 filename`.

Do not submit it by hand if you also submit electronically. That is a waste of paper. Fear not; I will be able to read your assignment if you put it on-line.