1 Memory Management

1.1 Segmentation

Consider the following segment table:

<table>
<thead>
<tr>
<th>Segment</th>
<th>Base</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>219</td>
<td>600</td>
</tr>
<tr>
<td>1</td>
<td>2300</td>
<td>14</td>
</tr>
<tr>
<td>2</td>
<td>90</td>
<td>100</td>
</tr>
<tr>
<td>3</td>
<td>1327</td>
<td>580</td>
</tr>
<tr>
<td>4</td>
<td>1952</td>
<td>96</td>
</tr>
</tbody>
</table>

What are the physical addresses for the following logical addresses?

a) 0,430  
b) 1,10  
c) 2,500  
d) 3,400  
e) 4,112

1.2 Paging

Consider a paging system with the page table stored in memory.

a) If a memory reference takes 200 nanoseconds, how long does a paged memory reference take if there is no TLB?

b) If we add a TLB and 75% of all page-table references are found in the TLB, what is the average memory reference time when a TLB access takes 2 nanoseconds?

c) A typical program has 20% memory instructions. Each instruction requires 1 cycle to execute and each memory operation in the cache takes 1 cycle. Assume there are 5% data TLB misses, each requiring 100 cycles (1 cycle access + 99 cycles overhead) to handle. Also, 10% of the data accesses are cache misses each of which takes 15 cycles (1 cycle access + 14 cycles overhead). How long would it take to execute 1000 instructions?
1.3 Virtual Memory

Consider a paged virtual address space composed of 32 pages of 2 KB each which is mapped into a 1 MB physical memory space.

a) What is the format of the logical address; i.e., which bits are the offset bits and which are the page number bits? Explain.

b) What is the length and width of the page table, disregarding the access right bits?

1.4 Page Replacement

Consider the following page access pattern:

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, assuming one, two, three, four, five, six, or seven frames?

a) LRU replacement

b) FIFO replacement

c) Optimal replacement