1. Square-memory implementations of large memories often support fast page-modes that allow a burst of consecutive memory reads from within the same row of the square memory.

a) How does this benefit performance on a cache miss?

b) An additional improvement allows the processor to specify a different ordering of the consecutive memory reads. For example, a processor might specify a burst of size 4 reads with offsets of 2-3-0-1 from the specified memory address. How does this benefit performance on a cache miss?

---

2. For the 8-bit data 01001011 develop the Hamming codeword for one-bit error detection and correction:

<table>
<thead>
<tr>
<th>C1</th>
<th>D1</th>
<th>D2</th>
<th>D3</th>
<th>D4</th>
<th>D5</th>
<th>D6</th>
<th>D7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Check bit C1 looks at bit positions 3, 5, 7, 9, and 11
Check bit C2 looks at bit positions 3, 6, 7, 10, and 11
Check bit C4 looks at bit positions 5, 6, 7, and 12
Check bit C8 looks at bit positions 9, 10, 11, and 12

b) If bit D5 gets flipped (an error), then how would we be able to detect an error?

c) If bit D5 gets flipped (an error), then how would we be able to know which bit to correct?

d) For the 8-bit data 11001001 develop the Hamming codeword for one-bit error detection and correction:

<table>
<thead>
<tr>
<th>C1</th>
<th>D1</th>
<th>D2</th>
<th>D3</th>
<th>D4</th>
<th>D5</th>
<th>D6</th>
<th>D7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Check bit C1 looks at bit positions 3, 5, 7, 9, and 11
Check bit C2 looks at bit positions 3, 6, 7, 10, and 11
Check bit C4 looks at bit positions 5, 6, 7, and 12
Check bit C8 looks at bit positions 9, 10, 11, and 12

b) If bit D5 gets flipped (an error), then how would we be able to detect an error?