Question 1:
Assume that the elements of the $n \times n$ matrix $A$ are stored row wise in memory (row major numbering) and that the cache size is enough to store only half the elements of $A$.

a) If the cache block size is 8 words, what is the cache miss rate while accessing the elements of $A$ during the following computation of the multiplication of $A$ by two vectors, $Y$ and $W$?

b) How can the computation be rearranged to improve the miss rate while accessing $A$?

a) The miss rate is 12.5%

b) Fuse the two loops to reduce the miss rate by half

```c
for (i = 0; i < n; i = i+1)
    { x[i] = 0;
      z[i] = 0;
      for (j = 0; j < n; j = j+1)
          { x[i] = x[i] + A[i][j] * Y[j];
            z[i] = z[i] + A[i][j] * W[j];
          };
    }
```

Question 2:
Assume a system with a 8KB cache and cache block size of 4 words. The following computation multiply $n \times n$ matrices, where $n = 40$ ($40 \times 40 \times 4 = 6.4$KB is enough to store all the elements of one $40 \times 40$ matrix).

a) What is the miss rate while accessing the elements of $B$ and $E$?

b) How can you rearrange the computation to improve this miss rate?

a) The miss rate is 25%

b) Splitting the loop reduces the miss rate by a factor of 40.

```c
for (i = 0; i < 40 ; i++)
    for (j = 0 ; j < 40 ; j++)
        { r = 0;
          v = 0;
          for (k = 0; k < 40 ; k++)
              { r = r + A[i][k]*B[k][j];
                v = v + D[i][k]*E[k][j];
              };
          C[i][j] = r;
          F[i][j] = v;
        };
```
for (i = 0; i < n; i++)
for (j = 0; j < n; j++)
{
    r = 0;
    v = 0;
    for (k = 0; k < n; k++)
    {
        r = r + A[i][k] * B[k][j];
        v = v + D[i][k] * E[k][j];
    }
    C[i][j] = r;
    F[i][j] = v;
}

for (i = 0; i < n; i++)
for (j = 0; j < n; j++)
{
    r = 0;
    for (k = 0; k < n; k++)
    r = r + A[i][k] * B[k][j];
    C[i][j] = r;
}

for (i = 0; i < n; i++)
for (j = 0; j < n; j++)
{
    v = 0;
    for (k = 0; k < n; k++)
    v = v + D[i][k] * E[k][j];
    F[i][j] = v;
}

b) Fuse the two loops to reduce the miss rate by half
**Question 3:**

b) What is the minimum Hamming distance between the code words corresponding to the five data words specified in (a)?

The distance between any two code words is at least 3

c) Assuming at most a single bit error, retrieve the correct data word corresponding to the code words: 1010101 and 1110111

- **1010101 → syndrome = 000 →**
  - no errors detected →
  - data word is 1101

- **1110111 → syndrome = 100 →**
  - error is in bit position 4 →
  - the corrected code word is 1111111 →
  - the decoded data word is 1111

<table>
<thead>
<tr>
<th>1</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>s1=0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>s2=0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>s3=0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**Syndrome = s3 s2 s1**

<table>
<thead>
<tr>
<th>1</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>s1=0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>s2=0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>s3=0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**Syndrome = s3 s2 s1**

**Question 3:**

**d) As described above, 1100 is encoded as 0111100. Assume that two bits are flipped while reading 0111100 from memory resulting in 1011100, what will be the outcome of the decoding process?**

- **1011100 → syndrome = 011 →**
  - error is in bit position 3 →
  - the corrected code word is 1001100 →
  - the decoded data word is 0100, which is not the original word 1100

<table>
<thead>
<tr>
<th>1</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>s1=1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>s2=1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>s3=0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Syndrome = s3 s2 s1**