CS455 - Intro to Computer Networks
Homework 2

- **Question 1 (20 pts)**
  - A **channel** has a **total bandwidth** of **1 Gbps** is **shared** using the **synchronous time division** technique
  - A **period** is divided into **640 time slots**.
  - **Question:**

    - What is the **smallest unit** of **bandwidth reservation** that you can make on this system ? (10 pts)

    - A **user** wants to **reserve 50 Mbps** transmission capacity on this **channel**

      How many time slots must we **reserve** to **provide** this **amount** of **bandwidth reservation** ? (10 pts)

- **Question 2 (20 pts)**
  - A **sender node** uses a **2 dimensional parity scheme** to transmit the following 4 **ASCII characters**:

```
1010000
0100101
0100010
1001001
```

    The **sender** uses ***even* parity** in **rows and** in **columns** and **transmits** the **bits** in a **row-wise** fashion (including the parity bit in each row).

    - Show the **series of *bits* transmitted** by the **sender node**. **(5 pts)**

      In other words: **first**, **encode** the above message, **then** show the **transmitted bits**.

  - A **receiving data link layer** uses the **2 dimensional parity scheme** with **even parity** for both columns and rows.

    Suppose the **receiver** receives the following **message**:

```
11110000
01010101
10101010
00001111
```

As you can see, all rows and columns have even parity. So the **receiver** will **accept** this **message without errors**.

Suppose the messages above that **was received in error**

Questions:

- Is it possible that the **received message** has **2 bits** that were **received** in **error** that the receiver **cannot** detect ? **(5 pts)**

  Answer:

  - Circle: **Yes** or **No**
  - If you have circled **yes**, then show an error pattern with **2 bit errors** by **circling the bits** that are **received** in **error** (and is ***undetectable*** by the **2-dim parity method**):

```
11110000
01010101
10101010
00001111
```

- Is it possible that the **received message** has **3 bits** that were **received** in **error** that the receiver **cannot** detect ? **(5 pts)**

  Answer:

  - Circle: **Yes** or **No**
  - If you have circled **yes**, then show an error pattern with **2 bit errors** by **circling the bits** that are **received** in **error** (and is ***undetectable*** by the **2-dim parity method**):

```
11110000
01010101
10101010
00001111
```

- Is it possible that the **received message** has **4 bits** that were **received** in **error** that the receiver **cannot** detect ? **(5 pts)**

  Answer:

  - Circle: **Yes** or **No**
  - If you have circled **yes**, then show an error pattern with **2 bit errors** by **circling the bits** that are **received** in **error** (and is ***undetectable*** by the **2-dim parity method**):

```
11110000
01010101
10101010
00001111
```

- **Question 3 (10 pts)**
  - A **sender** transmits the following **bit pattern** using the **Hammings code**:

```
                1111111
Bit position:   6543210987654321
                ---------------
Bit pattern:    0100000010000001
```

    I put the pattern under a lot of numbers to show the bit positions. You should read the numbers above the bit pattern as:

    ```
    bit position 1, 2, 3, 4, 5, 6, 7,8, 9, 10 , 11, 12, 13, 14, 15, 16
    ```

    from **right** to left. (The bit positions are written as 16, 15, 14, ..., 4,3,2,1 from **left** to right.)

  - **Show** the **bit pattern** that will be **transmitted**:

```
                2222222221111111111
Bit position:   9876543210987654321098765432l
                ----------------------------
Answer:                                      (line up bit position !)
```

    I.e.: **encode** the above message.

    NOTE: You **must line up** your answer with the **bit position** as I have done above to receive credit !

- **Question 4 (20 pts)**
  - **Note:**

    - This question is **not** related to **quesrion 3** above !

  - A **receiver** **received** the following **bit pattern** that is **encoded** using **Hamming code**:

```
Bit position:   6543210987654321
                ---------------
                0100000010000001
```

    Questions:

    - Which **one** bit will the receiver assume to be in error ? **(5 pts)**

    - What was the **original data** that was transmitted ? **(5 pts)**

```
Bit position: 6543210987654321
              ----------------
Answer:
```

Further clarification:

- I am asking what is the **original data *before*** the **Hamming code** was **applied**.

  You must use the **corrected Hamming code word** to answer this (the corrected Hamming code word is obtained from the previous question: Which one bit will the receiver assume to be in error ? - correct that bit and use the corrected Hamming code to obtain the original data).

NOTE: You **MUST** number your answer with the bit position as I have done above to receive credit !

- Show a **Hamming code word** in which **two bit errors** were made in the transmission that **results** in the **following bit pattern**: **(10 pts)**

```
Bit position: 6543210987654321
              ----------------
              0100000010000001
```

Circle in the **above figure** the **2 bits** that are **in error**.

Further clarification:

- You must find an **Hamming code** word `yyyyyyyyyyyyyyyy` such that:

```
Original Hamming code word: yyyyyyyyyyyyyyyy
After two bit errors:          x     x        <--- You MUST indicate WHICH
                               |     |             2 bits are in error.
                               V     V
Received by receiver:       0100000010000001
```

  The **difficulty** of this question lies in the fact that a **Hamming code** word **must** pass the **Hamming code test** (see: **click here**).

- **Question 5 (20 pts)**
  - A **sender** wants to send the following **message** **protected** by the **CRC polynomial 101**:

```
10000001
```

    What is the **bit pattern** that the **sender** will **transmit** ? **(10 pts)**

- A **receiver** using the **CRC polynomial 101** **receives** the following **bit pattern**:

```
10000001
```

  Will the **receiver** decide that the **message** was **correct** or **in error** ? Explain to get full credit. **(10 pts)**

- **Question 6 (10 pts)**
  - Show the **shift-register circuit** used to compute the **CRC code** using the **CRC polynomial 1100101** **(10 pts)**
