HW #3: Networking Questions
Spring 2020

Submit electronically as a PDF file called `hw3_netID.pdf` on Gradescope (see course website for due date)

Note: This assignment includes a written portion (this document) and a programming portion (separate document). Be sure to submit both!

1. **Bit Stuffing.**

a. A bit string, 10001111110100011111011, needs to be transmitted at the data link layer. What is the string transmitted across the Link after bit stuffing by the sender? Assume the same start/end flags as the ones used in class.

Frame payload: 10001111110100011111011
After bit stuffing: 01111110100011111**0**10100011111**0**01101111110

b. A frame is received by the data link layer, which was transmitted using bit stuffing: 011111101111101100011111011011111110. What is the bit string that the link layer passes up the stack to the network layer after bit de-stuffing?

A frame using bit stuffing: 0111111011111**0**1100011111**0**1101111110
After bit de-stuffing: 1111111000111111

2. **Hamming Code.**

a. Encode the message 10011011 to send.

i. Calculate the number of redundant bits:

$$2^r \geq 8 + r + 1 \quad \rightarrow \quad r = 4$$

ii. Insert redundant bits at positions of power of 2: 1, 2, 4, 8

P1	P2	D1	P4	D2	D3	D4	P8	D5	D6	D7	D8
0	**1**	1	**0**	0	0	1	**1**	1	0	1	1

iii. Calculate each redundant bit according to even parity:

$$P_1 = parity\ (3, 5, 7, 9, 11) = 0$$
$$P_2 = parity\ (3, 6, 7, 10, 11) = 1$$
$$P_4 = parity\ (5, 6, 7, 12) = 0$$
$$P_8 = parity\ (9, 10, 11, 12) = 1$$

Therefore, the redundant bits can be filled in the above table now.

b. What can be said about the correctness of the following received messages (Hint: Check for Hamming Code correctness using parity)?

i. 111000101011

The message has 12 bits, so there are 4 redundant bits for check.

$$C_1 = parity\ (1,3,5,7,9,11) = 1 \quad \rightarrow \quad wrong$$
$$C_2 = parity\ (2,3,6,7,10,11) = 0 \quad \rightarrow \quad correct$$
$$C_4 = parity\ (4,5,6,7,12) = 0 \quad \rightarrow \quad correct$$
$$C_8 = parity\ (8,9,10,11,12\ ) = 1 \quad \rightarrow \quad wrong$$

Because C_1 and C_4 check the same bit 9 and bit 11, but C_2 indicates that bit 11 is correct, then error happens on bit 9. Flip bit 9 from 1 to 0 so that message becomes 111000100011 which should be correct.

ii. 01110011011

The message has 11 bits, so there are 4 redundant bits for check.

$$C_1 = parity\ (1,3,5,7,9,11) = 1 \quad \rightarrow \quad wrong$$
$$C_2 = parity\ (2,3,6,7,10,11) = 1 \quad \rightarrow \quad wrong$$
$$C_4 = parity\ (4,5,6,7) = 0 \quad \rightarrow \quad correct$$
$$C_8 = parity\ (8,9,10,11\ ) = 1 \quad \rightarrow \quad wrong$$

Because C_1, C_2 and C_8 check the same bit 11. Flip bit 11 from 1 to 0 so that message becomes 01110011010 which should be correct.

3. **CRC Code.** Assume the $C(x) = x^4 + x^2 + 1$.

a. Encode the message 10110 with CRC.

The divisor polynomial C(x) is of degree 4, so we add 4 zeroes to the end of the message, which is 101100000. C(x) in binary form is 10101, the modulo-2 arithmetic operation is done as below:

									1	0	0	1	1
1	0	1	0	1	1	0	1	1	0	0	0	0	0
					1	0	1	0	1				
								1	1	0	0	0	
								1	0	1	0	1	
									1	1	0	1	0
									1	0	1	0	1
										1	1	1	1

The remainder is 1111, so the encoded message is 101101111.

b. What can be said about the correctness of the following received messages?

i. 110101110

									1	1	1	0	0
1	0	1	0	1	1	1	0	1	0	1	1	1	0
					1	0	1	0	1				
						1	1	1	1	1			
						1	0	1	0	1			
							1	0	1	0	1		
							1	0	1	0	1		
											0	1	0

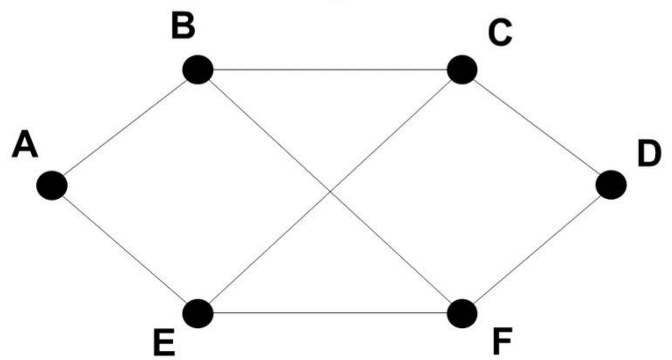
The remainder is not 0, so there must be some error in the received message.

ii. 110101100

									1	1	1	0	0
1	0	1	0	1	1	1	0	1	0	1	1	0	0
					1	0	1	0	1				
						1	1	1	1	1			
						1	0	1	0	1			
							1	0	1	0	1		
							1	0	1	0	1		
											0	0	0

The remainder is 0, so the received message is correct.

4. **Distance Vector Routing. Distance Vector Routing.** Consider the subnet shown below. Distance vector routing is used, and the following distance vectors have just come in to router C: **B**: (6, 0, 8, 10, 5, 5); from **D**: (4, 9, 7, 0, 8, 6); and from **E**: (7, 7, 4, 8, 0, 5). The measured distances/costs from C to **B**, **D**, and **E** are 5, 5, and 4, respectively. What will C's new routing table be after this update? Show both the outgoing router to use and the cost.



Routing Table Format:

Destination	Cost	Next Hop
A	9	D
B	5	B
C	0	C
D	5	D
E	4	E
F	9	E

5. **TCP Sequence Numbers.** To get around the problem of sequence numbers wrapping around while old TCP packets still exist, TCP could use 64-bit sequence numbers instead of 32 bits. However, theoretically, an optical fiber can run at 100 Terabits per second. What maximum packet lifetime would be required to prevent sequence number wrap-around even with 64-bit sequence numbers? Assume that each byte of a packet has its own sequence number (as TCP does).

   Sequence number is represented by 64 bits, which means there are 2^{64} different sequence numbers. The speed of byte transmission on an optical fiber is $\frac{100\times 2^{40}\ bits/sec}{8\ bits/byte} = 100 \times 2^{37} bytes/sec$. Since each byte of a packet has a sequence number, the maximum packet lifetime to prevent sequence number wrap-around should be:

$$\frac{2^{64}}{100 \times 2^{37}} sec = \frac{2^{27}}{100} \times \frac{1}{60} \times \frac{1}{60} h = 372.8h$$

6. **DNS.** Using an online whois lookup service like whois.net, look up duke.edu.

On what date was the domain registered? When does it expire? What are the DNS servers for this domain? Include a screenshot of your source.

The domain registered on 1986-06-02 and will expire at 2021-07-31T11:59:59Z.

DNS Servers:
DNS-AUTH-01.OIT.DUKE.EDU,
DNS-AUTH-02.OIT.DUKE.EDU,
DNS-NC1-01.OIT.DUKE.EDU

WHOIS LOOKUP

duke.edu is already registered*

Domain Name: DUKE.EDU
Registry Domain ID: 5059_DOMAIN_EDU-VRSN
Registrar WHOIS Server: whois.educause.net
Registrar URL: http://www.educause.edu/edudomain
Updated Date: 2018-06-08T13:57:29Z
Creation Date: 1986-06-02T04:00:00Z
Registry Expiry Date: 2021-07-31T11:59:59Z
Registrar: Educause
Registrar IANA ID: 365
Registrar Abuse Contact Email:
Registrar Abuse Contact Phone:
Domain Status: clientDeleteProhibited https://icann.org/epp#clientDeleteProhibited
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
Domain Status: clientUpdateProhibited https://icann.org/epp#clientUpdateProhibited
Name Server: DNS-AUTH-01.OIT.DUKE.EDU
Name Server: DNS-AUTH-02.OIT.DUKE.EDU
Name Server: DNS-NC1-01.OIT.DUKE.EDU
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of whois database: 2018-06-13T13:34:24Z <<<

7. **Internet Services.** Using netcat (the 'nc' command) in a terminal, manually display the following URL to the console.
   http://rabihyounes.com/awesome.txt

```
qx37@vcm-12680:~$ printf "GET http://rabihyounes.com/awesome.txt HTTP/1.1\r\nHost: rabihyounes.com\r\n\r\n" | nc rabihyounes.com 80
HTTP/1.1 200 OK
Date: Wed, 19 Feb 2020 01:04:23 GMT
Server: Apache
Upgrade: h2,h2c
Connection: Upgrade
Last-Modified: Fri, 08 Feb 2019 18:43:41 GMT
Accept-Ranges: bytes
Content-Length: 2360
Cache-Control: max-age=604800
Expires: Wed, 26 Feb 2020 01:04:23 GMT
Vary: Accept-Encoding
host-header: c2hhcmVkLmJsdWVob3N0LmNvbQ==
X-Endurance-Cache-Level: 4
Content-Type: text/plain
```