

CS447 – Networks and Data Communications

Summer 2026

Homework #3 Questions (**SOLUTIONS**)

Submission Due (each student is required to submit (by email as attached file) your solutions for this homework by **11:00AM on June 30**)

QUESTION #1

If a domain has 31,500 host computers (i.e., needs 31,500 IP addresses) and if the domain is supposed to have 20 subnets, what is its subnet mask (answer in the standard “x.x.x.x” format, where ‘x’ is a decimal number between 0 and 255)?

Assume that the 31,500 hosts are evenly distributed to the 20 subnets.

Show all your work for full credit.

Solution:

The number of the bits for the host addresses each subnet needs:

$$\text{ceiling}(\log_2((31,500/20))) = \text{ceiling}(\log_2(1575)) = 11 \text{ bits}$$

Since the subnet mask “masks” all the other bits other than the host address bits, the first $32-11 = 21$ bits should be ‘1’:

$$\begin{aligned} \text{subnet mask: } & 11111111.11111111.11111000.00000000 \\ & = \mathbf{255.255.248.0} \end{aligned}$$

QUESTION #2

Assuming the following parameters, find (calculate) (a) the host address (in the standard “x.x.x.x” format, where ‘x’ is a decimal number between 0 and 255) and (b) the subnet address of the given IP address (as an integer):

- the given IP address: 180.155.70.129/12
- subnet mask: 255.255.252.0000

Show all your work for full credit.

Solution:

- 180 = 10110100
- 155 = 10011011
- 70 = 01000110
- 129 = 10000001

- 252 = 11111100

The given IP: 1011 0100. 1001 1011. 0100 0110. 1000 0001

Subnet Mask: 1111 1111.1111 1111.1111 1100.0000 0000

(a) The host address

Subnet mask: 1111 1111.1111 1111.1111 1100.0000 0000

Inverted mask: 0000 0000.0000 0000.0000 0011.11111111

Apply the bitwise AND

Given IP: 1011 0100.1001 1011. 0100 0110. 1000 0001

Inverted mask: 0000 0000.0000 0000.0000 0011.1111 1111

Output: 00000000.00000000.00000010.10000001

= **0.0.2.129** (or **2.129**)

(b) The subnet address

Apply the bitwise AND

Given IP: 1011 0100.1001 1011.0100 0110.1000 0001

Subnet Mask: 1111 1111.1111 1111.1111 1100.0000 0000

Output: 1011 0100.1001 1011.0100 0100.0000 0000

Eliminate the first 12 bits (because of /12) and the last 10 bits (because of the '0' in the last 10 bits in the subnet mask):

1011 0100.1001 1011.0100 0100.0000 0000

= 1011.010001

$$= 1011.10001$$

$$= 101110001$$

$$= \mathbf{369}$$

Note: although “369” is the subnet address, any of the following is also acceptable this time:

- **1011.010001**
- **101110001**
- **11.136** (= 1011.10001000)