Problems

1. **[40 points]** In addition and amplification of what we have discussed thus far, I have attached to this examination some additional information that will explain some of the terminology used herein (e.g., URL). Consider a simplified model of a HTTP stream (not HTTPS because the material concerning security and encryption is still to come) within a TCP stream within an IPv4 stream within an IEEE 802.3 Frame stream. As you may recall, the 802.3 MAC uses a 48 bit address, and thus needs 6 hexadecimal digits to encode a NAC address. Call these $MAC_5 ... MAC_0$. IPv4 uses 4 hexadecimal digits to encode an IPv4 address, call these $V4_3 .. V4_0$. HTTP uses a URL, and assume that the URL has the format: http://some.dns.location.dns-top-level/some-way-to-find-information that one sees in a web browser. You will note that HTTP requires a port number (see: https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers)

Figure 1: Typical network data encapsulation from MAC to HTTP
From: https://tournasdimitrios1.files.wordpress.com/2011/01/networkpacket.jpg

1.1. For each of the layers, separately calculated, how many items (nodes) can be addressed, if applicable (e.g., there are source and/or destination numerical addresses)? Do not simply show an answer, but explain your reasoning (hint: combinatorics).

1.2. We have discussed functors as a formal, theoretical description between layers of a network. In terms of these layers, illustrate the functors between the layers, and explain how functors address the differences in topology and relevant information content at the different layers. You may show functors as composition "diagrams" such as

Figure 2: Composition diagram
From: https://en.wikipedia.org/wiki/File:Commutative_diagram_for_morphism.svg

1.3. Assume that the HTTP stream has 10 kbytes. How does this stream encapsulate in IPv4 packets? (Hint: each byte is one octet that is 8 bits.)

1.4. Taking the above result, or a made-up one of your choosing if you cannot calculate such a result, how does this data from HTTP encapsulating in IPv4 data then encapsulate in the 802.3 packet (or, in the nomenclature of the IEEE 802.3 standard, "frame")?

1.5. Assuming that the only information of interest is the payload of HTTP (what one might see on the screen using a web browser application such as Mozilla Firefox), what is the overall efficiency of the final encapsulated data stream in the IEEE 802.3 packet (again, "frame") for the HTTP data described above? That is, assuming, say, 100 octets of HTTP information to display, how many octets including all headers, etc., are required for those 100 octets to be received by the IEEE 802.3 network interface card (NIC), with the efficiency being the ratio between these? (In the interests of retention and graduation, if 100 octets were to be used but 1000 octets were ultimately received by the NIC to get those 100 octets, the efficiency would be 100/1000 or 0.10 .)

1.6. Assuming that only the information in the starting HTTP payload is the signal, and the rest of all packets (or frames) at all layers is noise with respect to this very restricted view of the information content of a channel, what is the effective Shannon-Hartley theorem relation between "channel capacity" and "bandwidth" for this specific datagram?

2. **[40 points]** Consider a set of intersecting rings as in the following figure. Here, a small square represents a node, including nodes that can transfer packets between rings, and each ring has an arrow that indicates the direction of packet flow on that ring. Each ring is labeled by a lower case Greek letter with the first ring labeled α .

   2.1. Display the adjacency matrix for the network in the figure.

   2.2. Which, if any, nodes are equivalent on the network and why? (Hint: Recall that equivalence in this context has to do with the number of links to a node as well as the information flow directions to and from a node. Thus if node A has two links to it, say link X into A and link Y leaving A, and node B has Y into B and link Z leaving B, then in this context, A and B are "equivalent". Note also that in other contexts, say one in which A has throughput capability α and B has throughput capability β, $\alpha \neq \beta$, then A and B in that context might not be equivalent. Note that α and β in this hint are not the same as the ring identification symbols in the set of intersecting rings that is a figure in this examination.)

   2.3. Which nodes, if any, represent single points of failure of the network?

   2.4. Display the weight matrix for the network.

   2.5. Using the Bellman-Ford algorithm, calculate a route from node S to node D on the network as displayed in Fig. 1. Show each step of the algorithm as you develop the route.

Figure 3: A set of intersecting rings

3. **[20 points]** Again refer to the above figure. Label each approximately at its center from the set $\{\alpha, \beta, \delta, \gamma, \epsilon\}$. Each ring represents an autonomous system (AS).

   3.1. What is meant by the term *autonomous sytem*?

   3.2. For the entire network from S to D, are there any single points of failure? If so, identify each both by the AS and the node number.

   3.3. Which nodes correspond to interior nodes for each AS?

   3.4. Assume that the maximum throughput of each segment is related to the metric (weight) associated with each segment as follows: $1 = 1$ Tbit$\cdot sec^{-1}$, $2 = 100$ Gbit$\cdot sec^{-1}$, $3 = 10$ Gbit$\cdot sec^{-1}$, $4 = 1$ Gbit$\cdot sec^{-1}$, $5 = 100$ Mbit$\cdot sec^{-1}$, etc. What is the maximum throughput from S to D; show your work and reasoning, not just a numerical answer.