



University  
of Glasgow

**Wednesday, 9 May 2018**  
**09:30 am - 11:00 am**  
**(1 hour 30 minutes)**

**DEGREES of MSc, MSci, MEng, BEng, BSc, MA and MA (Social Sciences)**

## **NETWORKED SYSTEMS (H)**

**Answer all 3 questions**

**This examination paper is worth a total of 60 marks.**

**The use of calculators is not permitted in this examination.**

**INSTRUCTIONS TO INVIGILATORS: Please collect all exam question papers and exam answer scripts and retain for school to collect. Candidates must not remove exam question papers.**

1. (a) The concept of *bridging* can be used to improve the scalability of an Ethernet by allowing several network segments to be connected together to form a larger network. Explain: 1) at what layer in the OSI reference model an Ethernet bridge operates; and 2) what layers of the OSI reference model describe the behaviour of single Ethernet links. [3]
- (b) To prevent forwarding loops, the Ethernet bridging protocol builds a spanning tree over the connected LAN segments. With the aid of an example, explain the operation of Perlman's distributed spanning tree algorithm, as used in the Ethernet protocol. [8]
- (c) A routing algorithm is used to find the best network layer path between two hosts that are not directly connected. Two types of routing algorithm are used: intra-domain routing and inter-domain routing. State what is the difference between these two types of routing, and describe in what environment each type of algorithm would be used. [4]
- (d) The Border Gateway Protocol (BGP) is used for inter-domain routing in the Internet. A BGP router builds its routing tables based on exchange of Autonomous System (AS)-path vectors giving routes to destination IP address prefixes. The routing information exchanged is often filtered to enforce policy, with the Gao-Rexford filtering rules being widely used. Describe the Gao-Rexford rules, and explain why they are desirable. [5]
2. (a) The two most widely used transport protocols in the Internet are TCP and UDP. Describe the service models provided by these two protocols, being sure to highlight their main differences. [8]
- (b) In the original design of the Internet, packet forwarding was based solely on the destination IP address of each packet, and the choice of transport protocol was solely a matter for the end systems. In practice, this is no longer the case, and it is difficult for end systems to use a transport protocol other than TCP or UDP. Explain what prevents the use of other transport protocols in the current Internet, and discuss the trade-offs around whether it is desirable to prevent the use of new transport protocols in this manner. State whether you think such blocking is overall helpful or harmful for the Internet, and justify your conclusion. [12]
3. (a) The second laboratory exercise for the course considered the development of a web server that uses the HTTP/1.1 protocol. The first part of this exercise involved correcting a bug where the provided web server code failed to correctly serve JPEG images. The fault occurred in the loop that reads the JPEG image from a file and then sends it to the client using the following code:

```

// Send the requested file
while ((rlen = read(inf, buf, BUFLen)) > 0) {
    if (send_response(fd, buf, strlen(buf)) == -1) {
        return -1;
    }
}

```

where `inf` is the file descriptor for the JPEG file, `fd` is the file descriptor for the socket connected to the client, and `buf` is a `char` array with `BUFLen` elements. The bug was corrected by changing the above code to read as follows:

```

// Send the requested file
while ((rlen = read(inf, buf, BUFLLEN)) > 0) {
    if (send_response(fd, buf, rlen) == -1) {
        return -1;
    }
}

```

Explain what is the effect of this change, and why it solves the problem with the application being unable to send JPEG images. [4]

- (b) Part two of the second laboratory exercise required you to modify the provided web server, so that if a browser requests a URL representing a directory, and the file `index.html` exists in that directory, the server returns a redirect (using the “307 Temporary Redirect” HTTP/1.1 response) asking the browser to fetch the `index.html` file instead.

Rather than issuing a redirect to the `index.html` file, an alternative design might be for the server to read the contents of the `index.html` file, if it exists, and return it to the browser in a “200 Ok” response when a URL representing a directory is requested. Discuss the advantages and disadvantages of these two designs. State which of the two designs you prefer, and justify your preference. [6]

- (c) In the web server code provided for the second laboratory exercise, the function `read_headers()` is used to read the HTTP/1.1 headers from the client socket. This function is written as follows:

```

static char *
read_headers(int fd)
{
    char    buf[BUFLLEN];
    char    *headers = malloc(1);
    size_t  headerLen = 0;
    ssize_t rlen;

    headers[0] = '\0';
    while (strstr(headers, "\r\n\r\n") == NULL) {
        rlen = recv(fd, buf, BUFLLEN, 0);
        if (rlen == 0) {
            // Connection closed by client
            free(headers);
            return NULL;
        } else if (rlen < 0) {
            free(headers);
            perror("Cannot read HTTP request");
            return NULL;
        } else {
            headerLen += (size_t) rlen;

```

```

        headers = realloc(headers, headerLen + 1);
        strncat(headers, buf, (size_t) rlen);
    }

    if (shutdown_requested) {
        printf("shutdown requested\n");
        return NULL;
    }
}

return headers;
}

```

where the parameter `fd` is the file descriptor for the socket representing the connection to the client, and `shutdown` is a global variable that indicates if the connection should be closed.

Discuss why the code is written in this manner, using a `while()` loop that wraps several calls to `recv()` and `realloc()` in a loop, rather than being just a single call to `recv()`. Explain how the function builds up complete HTTP/1.1 headers in memory. [10]