**CPSC 3600** section 001 Quiz #1

Spring 2021

Last update: 1/27/2021

**Take home quiz…you can use the book, your laptop, or any material. But you must**

**complete this individually.**

Name: **Anthony Jones**

Q1 After the following code runs, what is the value of \*a, \*b and c[0] displayed by the

printf ?

char c[8] ="1234567";

char \*a=NULL;

char \*b=NULL;

b=&c[5];

a=++b;

printf("\nquiz1: a:%c; b:%c; c[0]:%c\n",\*a,\*b, c[0]);

**\*a: 7 \*a is set to NULL, then set to \*( b+1 ), which is the index after b**

**\*b: 7 \*b is set to NULL, then incremented to \*( b+1 ), which is the same as \*a**

**c[0] 1 c[0] is the same as \*c, the first index of the array, which is 1**

Question 2 List the layers from the OSI seven layer network model that apply in a

TCP/IP network. Provide a 1sentence summary of each layer in a TCP/IP context.

The seven layers of the OSI network model are Physical, Datalink, Network, Transport, Session, Presentation, and Application. TCP/IP refers to the number of protocols and standards that have been developed by IETF:

**Physical:** This deals with the physical characteristics of hardware in networking. For example, the physical layer of TCP/IP might specify what pin connectors are used for network cables.

**Datalink:** This helps to identify the specific protocols being used in a network system. The specifications might include characteristics for WAN or LAN, or for some protocols like PPP (Point-to-Point) framing.

**Network:** This accepts and delivers packets for the network. This enables multiple data links to be combined into an internetwork, where multiple computer networks are connected so that they can exchange information. Most of the protocols here are routing protocols.

**Transport:** Responsible for dealing with transport protocols when sending data from one computer to another in a data link connection. This facilitates communication and provides an error-free interface for the network, while also allowing for a stable connection.

**Session:** A session is a “persistent logical linking of two software application processes, to allow them to exchange data over a prolonged period of time.” Basically, this layer deals with software application issues concerning setting up, managing, and ending sessions. This is often done through APIs, which are interfaces to manage these protocols. (<http://www.tcpipguide.com/free/t_SessionLayerLayer5.htm>)

**Presentation:** Deals with the presentation of data; specifically, protocols pertaining to data that is viewed within a network between two computers. Sometimes data needs to be managed in a way such that another computer is presented with an altered version of that data. Encryption, Compression, and Translation are all good examples; encryption obviously encrypts data, meaning the receiving user will get a version of the data that limits their access. Compression is when information is managed to fit in smaller file sizes, although usually decompression is required to properly access all the data. Translation is the process of sharing information between different types of computers, like PCs and UNIX.

**Application:** This layer provides interfaces and protocols needed by the users. This might include basic protocols like HTTP, the protocol for the web; or FTP, the protocol for transferring files between a client and a server. Each basic protocol has different strengths, weaknesses, uses, and applications; therefore, this layer is the most abstract and the least dependent for the OSI network model.

Question 3 In our git repo, get familiar with the CPPex1 code example. Perform an

experiment that runs the program 5 times, each time with 10000 iterations. Use the

helper scripts to automate running the experiment. Copy and paste the results.

**cpsc3600@vm1-ubuntu-1804:~/CPSC3600-Students/code/CPPCode/CPPex1$ ./go**

**./runEXPs.sh:Tue Feb 2 00:31:26 UTC 2021: 10000 1000000 1 5**

**0.347060576 10000 1.00**

**0.358420446 10000 1.00**

**0.349733118 10000 1.00**

**0.351298845 10000 1.00**

**0.346091720 10000 1.00**

**./runEXPs.sh:Tue Feb 2 00:31:28 UTC 2021: DONE !**

**cpsc3600@vm1-ubuntu-1804:~/CPSC3600-Students/code/CPPCode/CPPex1$**