**Explorations of Technology – Final Exam Review**

**Part I. Measuring Data**

***Bit*** - That smallest piece of data, represented as a single binary 0 or 1. Expressed with b (lower case)

***Byte*** – 8 Bits. It takes 1 byte to express a single letter or number on a computer. Expressed with B (upper case)

|  |  |  |
| --- | --- | --- |
| **Prefix** | **Symbol** | **Quantity** |
| Kilo | K | 1000 (thousand) |
| Mega | M | 1,000,000 (million) |
| Giga | G | 1,000,000,000 (billion) |
| Tera | T | 1,000,000,000,000 (trillion) |

You should be able to convert from one unit of measurement to another using the following four rules:

1. Moving down the table means you move the decimal point 3 places to the LEFT (divide by 1000).
2. Moving up the table means you move the decimal point 3 places to the RIGHT (multiply by 1000).
3. Converting from bits to bytes means you divide by 8.
4. Converting from byte to bits means you multiply by 8.

**Example# 1: 5.5 GB = MB**

We are going to use rule # 2, since we are moving up the table one spot. So we multiply 5.5 by 1000, giving us **5500 MB**.

**Example# 2: 48,000,000 Kb = GB**

In this example, we will use rules # 1, since we are moving down the table and rule #3, since we are converting from bits to bytes. Since we move down two rows (from kilo to giga), we must divide by 1000, twice, giving us 48 gb . Now we need to convert from gb to gB (bits to bytes). This requires us to divide by 8, giving us **6 GB**.

**Part II. Bandwidth and Data Transfer Times**

Bandwidth is the measurement of how fast we can move data. The standard measure is bits per second (bps). We can add the same prefixes above to more easily express faster data transfer speeds.

Ex: 20,000,000 bits per second = 20 Mbps (rule # 1, applied twice)

You should be able to calculate data transfer time, given a bandwidth and amount of data to transfer. Follow these steps to do this:

1. Get the bandwidth and data size into the same units.
2. Divide the data size by the bandwidth to get the number of seconds the transfer will take.
3. Convert seconds into minutes, hours, days. If there are more than 60 seconds, you should convert to minutes. If there are more than 60 minutes, you should convert to hours. If there are more than 24 hours, you should convert to days. Use this table to help:

|  |  |  |
| --- | --- | --- |
| **Conversion** | **Math** | **Example** |
| Seconds to Minutes | Divide seconds by 60 | 2400 seconds / 60 = 40 minutes |
| Minutes to Hours | Divide minutes by 60 | 420 minutes / 60 = 7 Hours |
| Hours to Days | Divide hours by 24 | 72 hours / 24 = 3 Days |

Example # 1: You want to download a 4.7 GB DVD image. Your home internet speed is 20 Mbps. How long will this take?

**Step # 1** – Convert 4.7 GB into Mb. Using rules # 2 and # 4. 4.7 \*1000 \* 8 = **37600 MB**

**Step # 2** – 37600 Mb / 20 Mbps = **1880 seconds**

**Step # 3** – 1880 seconds / 60 = **31.333 minutes** Answer

Example # 2 – You want to copy a 2.2 TB file onto a portable hard drive. The drive can copy data at 480 Mbps. How long will this take?

**Step # 1** – Convert 2.2 TB to Mb. Again, using rules #2 (twice) and #4 - 2.2 \* 1000\*1000\*8 = **17,600,000 Mb**

**Step # 2**  - 17,600,000 Mb/ 480 Mbps = **36,667 seconds**

**Step # 3** – 36,667 / 60 = 611 minutes 611 minutes/60 = **10.19 Hours** Answer

**Part III. Binary/Hexadecimal/Decimal Conversions**

For the exam, you should be able to convert between any of these number systems.

Binary – You will be given an 8 digit binary number, built with only 1s and 0s. Each of the 8 numbers represents a different value in decimal, our number system. With 8 binary digits, you can create decimal numbers in the range from 0 to 255.

128 64 32 16 8 4 2 1

**Conversion # 1** – Binary to Decimal: Put 8 binary digits under the 8 values above. Add the #s that have a 1 under them.

Ex: Convert 10001101 to decimal.

128 64 32 16 8 4 2 1

1 0 0 0 1 1 0 1

In this example, the 1s fall under the 128, 8, 4 and 1 columns, so we simply add those numbers together

128 + 8 + 4 + 1 = 141, so binary 10001101 = **141** (decimal.)

**Conversion # 2** – Decimal to Binary: Starting with 128, subtract the 8 values from the decimal, as long as result is not negative. If you can do it, put a 1 in that column. Should hit 0 in the end if done correctly.

Ex: Convert 189 to binary

* We can subtract 128 from 189, so we put a 1 in the 128 column and have 61 remaining (189-128)
* We cannot subtract 64 from 61 (We would go negative), so we put a 0 in the 64 column
* We can subtract 32 from 61, so we put a 1 in the 32 column and have 29 left.
* We can subtract 16 from 29, so we put a 1 in the 16 column and have 13 left.
* We can subtract 8 from 13, so we put a 1 in the 8 column and have 5 left.
* We can subtract 4 from 5, so we put a 1 in the 4 column and have 1 left.
* We cannot subtract 2 from 1, so we put a 0 in the 2 column.
* We can subtract 1 from 1, so we put a 1 in the 1 column and have 0 left.

The answer is **10111101**

Hexadecimal is a common number system used in the IT field. It has 16 different symbols used to express numbers. It uses 0 to 9, as well as the letters A through F. The value of each letter is as follows:

A = 10 B = 11 C = 12 D = 13 E = 14 F = 15

**Conversion # 3** – Binary to Hex: The conversion from binary to hex is easy. The rule to remember is that 4 binary digits = 1 hex digit. Since we give binary in 8 bit chunks, they can always be converted into 2 hex digits. The key is to split the binary in 4 bit chunks. Convert each chunk to a value 0 to 15. If the result is greater than 9, be sure to use the above conversions to express the letter A through F.

Ex: Convert 10100101 to Hex.

Split the binary in 4 bit chunks: 1010 and 0101

Convert each 4 bit chunk separately using: 8 4 2 1 as the column place values

**8 4 2 1 8 4 2 1**

1 0 1 0 0 1 0 1

8 + 2 = 10 4 + 1 = 5

10=A 5

Answer: 10100101 = **A5** (hex)

**Conversion # 4** – Hex to Binary: Another easy conversion. Treat each hex digit as a number 0 to 15. Convert each number to a 4 digit binary number. Glue to two together to get your final answer:

Ex: Convert D9 to binary.

D = 13 9

8 4 2 1 8 4 2 1

1 1 0 1 1 0 0 1

Answer: 11011001 = D9

**Conversion # 5** – Decimal to Hex: Use conversion # 2 to get from decimal to binary, then conversion # 4 to get from binary to hex.

Ex: Convert 203 to Hex.

203 decimal converts to binary:

128 64 32 16 8 4 2 1

1 1 0 0 1 0 1 1

11001011 binary converts to hex:

8 4 2 1 8 4 2 1

1 1 0 0 1 0 1 1

12 11

C B

So 203 decimal = **CB** (hex)

**Conversion # 6** – Hex to Decimal: Use conversion # 4 to get from hex to binary, then conversion # 1 to get from binary to decimal.

Ex: Convert 8F to decimal.

8F converts to binary:

8 F = 15

1000 1111

Then convert 10001111 to decimal

128 64 32 16 8 4 2 1

1 0 0 0 1 1 1 1

128+8+4+2+1 = 143

8F = **143** (decimal)

**Part IV. Parts of the Computer**

You should able to identify, from images, the following computer parts. You should also be able to explain the primary purpose of each, as well as determine which components connect to one another.

CPU

* Central Processing Unit
* Brains of the computer
* All decisions are made by the CPU
* Must match the socket of the motherboard to fit.
* Coated with ***thermal paste*** to conduct heat better
* A ***heat sink*** rests atop the CPU to move the heat off the CPU
* A ***fan*** is usually placed on top/near the heat sink to remove the heat from the computer system.

RAM

* Random Access Memory
* Stores the programs and data you are currently using.
* Disappears when the computer is turned off.
* Fits into DIMM slots on the motherboard.
* Usually measured in GB

Motherboard

* Printed circuit board that is using to connect all other components together.
* Make sure you match the motherboard socket with the CPU ***socket***.
* Need cables to connect drives to motherboard
* Current motherboards have various components built into it: Sound, Graphics, Ethernet port, USB ports

Power Supply

* Amount of power is measure in watts.
* This provides power to all components.
* Directly connects to motherboard, graphics cards and drives.
* All other components get their power through the motherboard power connection.

Hard Drive

* Permanently stores data.
* Data remains, even when the computer is turned off.
* Usually measured in GB or TB.

Case

* Metal chassis that protects the internal parts of the computer.
* Cases come in different size and must match the motherboard, power supply and drives that you intend to put in it.

Optical Drive

* 3 different generations of Optical Drives

|  |  |  |
| --- | --- | --- |
| **Generation** | **Storage Capacity** | **Base Read/Write Speed (1X)** |
| CD-ROM | 700 MB | 150 KBps |
| DVD | 4.7 GB | 1.3 MBps |
| Blu-Ray | 25 GB | 4.3 MBps |

* Uses X (multiplier) terminology to express how much faster than the base read/write speeds the drive can transfer data
* Be able to calculate Read/Write times giving the technology and multiplier. Similar to bandwidth problems.

Ex: You have a 5X Blu-ray burner. You want to write 10 GB of data onto a Blu-ray disk. How long will it take?

* Base write speed on a Blu-ray is 4.3 MBps. This is five times faster than the base, so 4.3 \* 5 = 21.5 MBps
* Converting 10 GB to MB means multiplying by 1000, so 10,000 MB.
* 10,000 MB / 21.5 MBps = 465 seconds
* 465 seconds / 60 = **7.75 minutes**

**Part V. Hacking**

* You should be able to identify from a description the 12 following types of hacks:

Brute Force Password Guessing Phishing Fake Website

Keylogger Trojan Horse Back Door Attack DOS Attack

Smurfing Packet Sniffing Fake Access Point Cookie Theft

* You should understand the difference between a black hat and white hat hacker.

**Part VI. Console Commands**

* You should understand how a batch file works
* You should know how to get to the command line interface (CLI) – **CMD**
* You should know the purpose and syntax for the following console commands:

CLS COLOR DIR CD

MD RD DEL TYPE

REN COPY ECHO C:

**Part VIII. HTML**

* You should know what the following HTML tags do and the syntax to use them:

<HTML> <TITLE> <BODY> <B>, <I>, <U>

<P> <BR> <SUB>, <SUP> <OL>, <UL>, <LI>

<IMG SRC> <A HREF> <CENTER> <TABLE>, <TR>, <TD>

<H1> to <H6> <FONT>

* You should be able to use table to create a webpage layout
* You should be able to use the following attributes to change the appearance of different tags:

WIDTH HEIGHT BGCOLOR BORDER

ALIGN COLSPAN (Tables) ROWSPAN (Tables)

**Part IX. Javascript Drawing Commands**

You should be familiar with the following Javascript drawing commands and be able to use them to create simple drawings:

* rect (x, y, w, h) – Draws a rectangle w pixels wide and h pixels tall, with its upper left corner location at (x, y)
* ellipse (x, y, w, g) – Draws an ellipse (circle or oval) w pixels wide and h pixels tall, centered around (x, y)
* triangle(x1, y1, x2, y2, x3, y3) – Draws a triangle using the three points indicated: (x1, y1), (x2, y2) and (x3, y3)
* line(x1, y1, x2, y2) – Draws a line from (x1, y1) to (x2, y2)
* background (r, g, b) – Uses values from 0 to 255 to control the red, green and blue concentration.
* Fill(r, g, b) – Sets fill color for shapes
* Stroke(r, g, b) – Sets the outline color for shapes
* strokeWeight(thickness) – Changes the thickness of lines and outlines
* noStroke() – Turns off outlines for shapes
* arc(x, y, w, h, start, stop) – Same as ellipse, but add the starting angle and ending angle for the arc. See below

270

180 0

90