

# Lab 1

Madiba Hudson-Quansah

## Part 1

### Question 1

Can you name the main components of a desktop computer discussed and briefly explain their functionalities?

#### **Solution:**

**CPU** - The Central Processing Unit. The brain of the computer, responsible for executing instructions.

**Motherboard** - The main circuit board of the computer, which connects all the components together allowing them to communicate.

**RAM** - Random Access Memory

**Hard Drive** - The storage device of the computer, where all the data is stored.

**Power Supply** - Supplies power to the computer.

**Graphics Card** - Responsible for rendering images on the screen.

### Question 2

How do the components interact with each other to enable the computer to function?

**Solution:** Through electrical signals transferred through the motherboard.

### Question 3

Differentiate between RAM (Random Access Memory) and storage devices (HDD/SSD). What are their respective functions?

**Solution:** RAM is used to store data that is currently being used by the computer for example running programs. RAM is volatile meaning when the computer is powered off the data stored in RAM is lost. Storage devices are used to keep data that needs to be available long term even if the computer is powered off.

### Question 4

How does the speed and capacity of memory and storage impact overall system performance?

**Solution:** The speed of memory and storage directly correlates to the speed of read and write operations of the computer. The faster the memory and storage the faster the computer can read and write data. The capacity of memory and storage determines how much data can be stored and accessed by the computer and has an inverse relationship with speed, i.e. the more data that can be stored by a device the slower its read and write speeds are, this is called the memory hierarchy.

### Question 5

In what scenarios is a powerful GPU particularly beneficial?

**Solution:** GPUs are useful in situations where graphics processing is required such as gaming, video processing and rendering, and video editing. GPUs are also useful in processing vector and matrix operations which are common in machine learning and other scientific computing applications. This is due to the SIMD (Single Instruction Multiple Data) and MIMD (Multiple Instruction Multiple Data) architecture of GPUs which allows them to process multiple data points in parallel.

### Question 6

How does the power rating of the Power Supply Unit (PSU) influence the system's capabilities?

**Solution:** The power rating of the PSU determines how much power the computer can draw from the wall. The power rating of the PSU should be higher than the power consumption of the computer to ensure that the computer can run at

full capacity. If the power rating of the PSU is lower than the power consumption of the computer the computer will not be able to run at full capacity and may not even turn on.

#### Question 7

Compare Solid State Drives (SSD) and Hard Disk Drives (HDD) in terms of materials and manufacturing, energy consumption and recyclability. Which one would you prefer when purchasing your next laptop and why?

**Solution:** SSDs use flash memory to store data while HDDs use spinning disks. This means that SSDs are faster and more reliable than HDDs but are more expensive to manufacture therefore cost more. This also means SSDs are more resistant to physical damage such as drops and are more energy efficient than HDDs. SSDs are also more recyclable than HDDs as they do not contain any moving parts. I currently use an SSD in my laptop and would prefer to use an SSD in my next laptop as I value speed and reliability over cost.

#### Question 8

Why do you need a cooling unit/heat sink?

**Solution:** We heat away from the CPU and other components of the computer to prevent them from overheating. Overheating can cause the computer to slow down or even shut down. Heat sinks are used to increase the surface area of the component to allow more heat to be dissipated.

## Part 2

### Question 9

How many hours do you typically use or keep your PC running each day? State any assumptions you make.

**Solution:** Around 17 hours as I regularly use my machine for taking notes in and out of class which accounts for most of the day but I also always keep my machine in hibernation mode on my Linux dual boot so I don't count that time.

### Question 10

Based on your PC specifications, write a Python script(s) for the computation of the energy consumption (in kWh) for the lower and upper boundaries of the following PC components:

1. CPU
2. RAM
3. Motherboard
4. HDD/SSD

**Solution:**

### Question 11

3. According to the Electricity Company of Ghana (ECG), the amount of electricity you consume determines the lifeline bracket you fall into. The different lifeline brackets and costs are:

- 0 – 50kWh: GHC 0.34
- 51 – 300kWh: GHC 0.67
- 301 – 600kWh: GHC 0.87
- 601kWh and above: GHC 0.97

Now that you have measured the total energy consumption of your PC, calculate your electricity bill for the month of January (with a Python script(s)) when your graphics card is in:

1. Load power mode
2. Idle power mode

**Solution:**

1. Bill (Lower Limit): GHC 57.58  
Bill (Upper Limit): GHC 90.63
2. Bill (Lower Limit): GHC 21.56  
Bill (Upper Limit): GHC 37.31

### Question 12

What lessons or insights have you learnt from the activity?

**Solution:** I have learnt that the power consumption of a computer can be quite high and that it is important to consider it when purchasing a computer, and even more so when building a computer. This also suggests that a large amount of energy is used by computers as this is just my machine and there are millions of other machines in the world.

## Part 3: Logisim

### Part 1

#### Question 13

Create new circuits and illustrate the following gates with 2 inputs:

1. EXOR
2. EXNOR
3. NAND
4. NOR

Draw their truth tables and verify the behavior of the corresponding circuit.

**Solution:**

1.	$x$	$y$	$x \oplus y$
	0	0	0
	0	1	1
	1	0	1
	1	1	0

2.	$x$	$y$	$\overline{x \oplus y}$
	0	0	1
	0	1	0
	1	0	0
	1	1	1

3.	$x$	$y$	$\overline{x \cdot y}$
	0	0	1
	0	1	1
	1	0	1
	1	1	0

4.	$x$	$y$	$\overline{x + y}$
	0	0	1
	0	1	0
	1	0	0
	1	1	0

### Part 2: Boolean Formulas and Truth Tables

#### Question 14

Every Boolean function has three equivalent representations: a Boolean formula, a Boolean circuit, and a truth table. A straightforward example of a Boolean function is the majority function. Given  $n$  inputs, the majority function is true if more than half the inputs are true; otherwise, it is false. Here is the Boolean formula for the three-input version of the majority function:

$$(A \cdot B) + (B \cdot C) + (A \cdot C)$$

Recall that the  $\cdot$  symbol means AND, and the  $+$  symbol means OR. The parentheses are used to specify the order of operations, just like in arithmetic. Confirm that the Boolean formula above is correct by completing the truth table below. Make sure that the final column matches the behavior of the majority function.

**Solution:**

A	B	C	$(A \cdot B)$	$(B \cdot C)$	$(A \cdot C)$	$(A \cdot B) + (B \cdot C) + (A \cdot C)$
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
0	1	1	0	1	0	1
1	0	0	0	0	0	0
1	0	1	0	0	1	1
1	1	0	1	0	0	1
1	1	1	1	1	1	1

## Part 4: Securing the Safe

### Question 15

As an electronic engineer, you are required to design and implement a combinational logic circuit in Logisim-Evolution that solves the problem of securing the safe containing the class resources. The safe should be controlled by three (3) people, the class Representative (CR), the Faculty Intern (FI), and the Course Lecturer (CL). The safe should be accessible if the course lecturer is present, indicated as 1. In the absence of the course lecturer, the safe should only be accessible if both the FI and the Class Representative are present.

Construct a truth table, find the SOP implementation for the truth table, and design the corresponding circuit. Make sure to test your circuit and compare it with your truth table.

**Solution:** Let  $x$  be the course lecturer,  $y$  be the faculty intern, and  $z$  be the class representative. Let  $F$  be the output of the circuit

$x$	$y$	$z$	$(y \cdot z)$	$F$
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$F(x, y, z) = \bar{x}yz + x\bar{y}\bar{z} + x\bar{y}z + xy\bar{z} + xyz$$

## Part 5: Reflection Questions

### Question 16

How does an AND gate function, and what are its primary applications?

**Solution:** An AND gate only produces an output when both of its inputs are powered/true. It is mainly used in digital adders and condition control.

### Question 17

Describe the operation of an OR gate and provide examples of where it is commonly used.

**Solution:** An OR gate produces an output when at least one of its inputs is powered/true. It is commonly used in combining signals and data selection

### Question 18

What is the purpose of a NOT gate, and how does it alter its input signal?

**Solution:** A NOT gate inverts its input signal, i.e. if the input signal is true the output signal is false and vice versa. It alters the input signal by inverting it.

### Question 19

Explain the significance of XOR gates in digital systems, providing examples where they are indispensable.

**Solution:** XOR (Exclusive OR) gates only output true when only one of their inputs is true. They are used in half-adders and full-adders.

### Question 20

How do NAND and NOR gates differ from their counterparts, and why are they referred to as universal gates?

**Solution:** NAND and NOR gates are the inverse of AND and OR gates respectively, i.e. they are basically the original gates output passed through a NOT gate this is where they differ from their counterparts. They are called universal gates because it is possible to construct any other logical gate using only NAND or NOR gates. For example an AND gate can be constructed by passing both outputs of one NAND gate through another NAND gate, as shown in the truth table below, where  $N_1$  is the output of the first NAND gates and  $N_2$  is the final output.

$x$	$y$	$N_1 = \overline{x \cdot y}$	$N_2 = \overline{N_1 \cdot N_1}$
0	0	1	0
0	1	1	0
1	0	1	0
1	1	0	1