Question 1. **What is Assembly Language? As a Computer Engineer why do you need to know Assembly Language? Do different processors/computer architectures have different assembly languages?**

- Assembly Language is a low-level programming language made for certain types of processors that represents various instructions in symbolic code. Computer Engineers should know Assembly Language because it is a time and work efficient programming language to use. There are no different assembly languages

Question 2. **Describe the Fetch-Decode-Execute cycle of the processor.**

- The Fetch-Decode-Execute cycle is the basic operation instruction cycle of a computer. The "fetch" stage gets program commands from the computer's memory. The "decode" stage makes sure the computer understands the program by deciphering it. The "execute" stage carries out the requests.

Question 3. **Do you think your laptop processor and your cellphone processor go through this fetch-decode-execute cycle?**

- Yes

Question 4. **What are registers? Why do the processors need registers?**

- Registers are internal memory storage locations. Processors need registers because it speeds up processor operations. Storing data in the main memory slows down the processor so storing data elements in registers for processing prevent that from happening.

*Question 5***. Name all the registers in 8086 Architecture? Write down their function in short.**

- AX (Accumulator): Input/Output and arithmetic instructions
- BX (Base register): Indexed addressing
- CX (Counter): Store loop count in iterative operations
- DX (Data register): Input/Output operations
- SP (Stack Pointer): Points to the topmost item of the stack
- BP (Base Pointer): Accesses parameters passed by the stack
- SI (Source Index): Pointer addresses data and used as a source for some string operations
- DI (Destination Index): Pointer addresses data and used as a destination for some string operations