Practice Questions
Computer Organization and Architecture IT 3005

Basic Operation of a Computer

1. **What is the difference between MAR and MDR?**
2. **Why do we need a Program Counter (PC) in a computer system? 3.**
3. **What is the role of the Instruction Register (IR)?**
4. **If an instruction says ADD R1, R2 and both registers are inside the processor, why don't we need to access memory?**

   Hint: Consider where R1 and R2 are located.

5. **Assume a 64-bit machine. What will be the increment value of the Program Counter after fetching one instruction?**
6. **If an instruction is stored at memory location 1000 and each instruction takes 4 bytes, what will be the address of the next instruction?**
7. **Why do we need both an address bus and a data bus? Why can't we use just one?**

   Hint: Think about what information each bus carries.

8. **What are the possible disadvantages of a single-bus architecture in a modern multi-core system?**
9. **In the instruction ADD R1, LOCA, why are two memory reads required? Could it be optimized further?**
10. **How does the use of multi-bus architecture improve performance compared to a single bus system?**
11. **Trace the steps your computer will take to execute MOV R3, [4500]. What registers will be used?**
12. **Imagine the PC is pointing to memory location 204. The instruction at 204 is ADD R2, 1000. What are the micro-operations involved in fetching and executing this instruction?**
13. **Can the content of MDR and IR ever be the same during the execution of an instruction?**

    Hint: Think about instruction fetch and data read.

14. **If a system has a 12-bit address bus, how many memory locations can it address?**
15. **How would execution change if both operands of an instruction were in memory instead of registers?**

    Hint: Consider how many memory accesses would be required.

Architecture vs Organization & Modern Trends

Part A – Short Answer (Concept Check)

1. Define computer architecture and computer organization in your own words.
2. What is backward compatibility? Give one example from processor families.
3. In Intel's Core i5 and Core i7 processors, the architecture is the same but organization differs. Explain how.

   Hint: Consider core count, cache, speed, and other internal resources.
4. State two key differences between ARM and x86 architectures.
5. What does heterogeneous architecture mean? Give an example.

Hint: Think of processors with two types of cores doing different jobs.

Part B – Comparative / Analytical

6. Compare Pentium Pro and Celeron in terms of architecture and organization.
7. Explain how Big. LITTLE architecture improves energy efficiency.
   Hint: Why not always use the fastest cores?
8. Discuss why modern laptops are increasingly shifting from x86 to ARM.
   Hint: Think battery life, heat, and performance balance.
9. Describe how parallelism is used to improve performance in modern processors.
   Hint: More than one core or thread working at the same time.
10. Explain why an application written for 80186 runs on 80286 but not always the other way round.

Part C – Application / Critical Thinking

Design Thinking

11. If you were making a new mobile phone processor, how would you make it **fast** but also **save battery**?
    Hint: Think about mixing different types of cores, controlling speed, and using modern chip technology.
12. **Core Usage Examples**
13. Give one example of a task where **big, fast cores** are better.
14. Give one example of a task where **small, low-power cores** are better.
15. **Many Small Cores + Few Big Cores**
16. If a CPU had **100 small cores** and **4 big cores**, what kind of jobs would it be best at?
    Hint: Think about jobs that can be split into many small parts vs jobs that need one fast core.
17. **Impact of Core Changes**
18. If Apple made a processor with **more big cores** but **fewer small cores**, what would happen to **speed** and **battery life**?
19. If your computer is doing **video rendering** and **light web browsing** at the same time, which tasks should go to big cores and which to small cores? Why?