**Problem 1.1:**  
**Aside from the smart cell phones used by a billion people, list and describe four other types of computers.**1. Desktops: Traditional personal computers that are used for general-purpose tasks like word processing, web browsing, gaming, and productivity applications.  
2. Laptops: Portable computers, used for a wide variety of tasks like desktops but with the added benefit of mobility.  
3. Servers: Computers designed to manage network resources, including web servers, file servers, and database servers.  
4. Embedded Systems: Computers integrated into other devices (e.g., microwaves, car engines, or industrial machines) to perform dedicated functions.

**Problem 1.2:  
The eight great ideas in computer architecture are similar to ideas from other fields. Match the eight ideas from computer architecture, “Design for Moore’s Law”, “Use Abstraction to Simplify Design”, “Make the Common Case Fast”, “Performance via Parallelism”, “Performance via Pipelining”, Performance via Prediction”, “Hierarchy of Memories”, and “Dependability via Redundancy” to the following ideas from other fields:**

1. **Assembly lines in automobile manufacturing**Performance via Pipelining. Much like assembly lines in manufacturing, where different stages work concurrently to improve efficiency.
2. **Suspension bridge cables**Dependability via Redundancy. Ensuring stability even if some cables fail, like redundancy in computer systems to enhance reliability.
3. **Aircraft and marine navigation systems that incorporate wind information**Performance via Prediction. These systems use environmental data to predict the best course of action, much like branch prediction in processors.
4. **Express elevators in buildings**Make the Common Case Fast. Express elevators skip floors, focusing on the most common and important floors to increase efficiency.
5. **Library reserve desk**Hierarchy of Memories. Reserving at the desk is mimicking how faster memory (like caches) is used for frequently accessed data in computing.
6. **Increasing the gate area on a CMOS transistor to decrease its switching time**Design for Moore’s Law. By increasing the size of transistors, the switching speed improves, which reflects how computer architectures are designed to anticipate advances in transistor technology.
7. **Adding electromagnetic aircraft catapults (which are electrically-powered as opposed to current steam-powered models), allowed by the increased power generation offered by the new reactor technology**Performance via Parallelism. More powerful reactors allow the addition of more energy-intensive systems, just like parallelism uses multiple processors or cores to improve computational performance.
8. **Building self-driving cars whose control systems partially rely on existing sensor systems already installed into the base vehicle, such as lane departure systems and smart cruise control systems**Abstraction to Simplify Design. Self-driving cars rely on abstractions (existing sensors) to simplify the design and development process, just as computer architectures use abstractions to simplify complex system designs.

**Problem 1.5:  
Consider three different processors P1, P2, and P3 executing the same instruction set. P1 has a 3 GHz clock rate and a CPI of 1.5. P2 has a 2.5 GHz clock rate and a CPI of 1.0. P3 has a 4.0 GHz clock rate and has a CPI of 2.2.**

Given the following data:  
**P1**: Clock rate = 3 GHz, CPI = 1.5  
**P2**: Clock rate = 2.5 GHz, CPI = 1.0  
**P3**: Clock rate = 4.0 GHz, CPI = 2.2

1. **Which processor has the highest performance expressed in instructions per second?**Formula to find Performance  
   Performance = Clock Rate / CPI  
     
   P1: 3x109 / 1.5 = 2.0 × 109 instructions per second  
   P2: 2.5×109 / 1.0 = 2.5 × 109 instructions per second  
   P3: 4×109 / 2.2 = 1.818 × 109 instructions per second
2. **If the processors each execute a program in 10 seconds, find the number of cycles and the number of instructions.**

P1: Execution time = 10 seconds, Clock rate = 3GHz, CPI = 1.5  
 Number of cycles = 3 × 109 × 10 = 30 × 109 cycles  
 Number of instructions = 30 × 109 / 1.5 = 20 × 109 instructions

P2: Execution time = 10 seconds, Clock rate = 2.5GHz, CPI = 1.0  
 Number of cycles = 2.5 × 109 × 10 = 25 × 109  
 Number of instructions = 25 × 109 / 10 = 25 × 109 instructions

P3: Execution time = 10 seconds, Clock rate = 4.0GHz, CPI = 2.2  
Number of cycles = 4.0 × 109 × 10 = 40 × 109 cycles  
 Number of instructions = 40 × 109 / 2.2 = ~18.18 × 109 instructions

**Problem 1.6:  
Consider two different implementations of the same instruction set architecture. The instructions can be divided into four classes according to their CPI (class A, B, C, and D). P1 with a clock rate of 2.5 GHz and CPIs of 1, 2, 3, and 3, and P2 with a clock rate of 3 GHz and CPIs of 2, 2, 2, and 2.**

**Given a program with a dynamic instruction count of 1.0E6 instructions divided into classes as follows: 10% class A, 20% class B, 50% class C, and 20% class D, which implementation is faster?**

P1: Clock rate = 2.5 GHz, CPIs for classes A, B, C, D = 1, 2, 3, 3.  
P2: Clock rate = 3 GHz, CPIs for classes A, B, C, D = 2, 2, 2, 2.  
Instruction mix: 10% class A, 20% class B, 50% class C, 20% class D.

1. **What is the global CPI for each implementation?**

P1: Global CPI = (0.1×1) + (0.2×2) + (0.5×3) + (0.2×3) = 0.1 + 0.4 + 1.5 + 0.6 = 2.6  
P2: Global CPI = (0.1×2) + (0.2×2) + (0.5×2) + (0.2×2) = 0.2 + 0.4 + 1.0 + 0.4 = 2.0

1. **Find the clock cycles required in both cases.**

P1: Clock cycles = 2.6 × 1 × 106 = 2.6 × 106 cycles  
P2: Clock cycles = 2.0 × 1 × 106 = 2.0 × 106 cycles  
  
Conclusion  
P2 is faster due to fewer clock cycles required