1. What is the **name** for a computer architecture in which the **main memory holds** not only the data to be used but also **the program instructions to be executed**?

   the Von Neumann Architecture

2. **Order** the following architectural layers from **lowest to highest**:

Assembly Language	**Application Program**	(highest)
Operating System	**Problem Oriented Language**	
Application Program	**Assembly Language**	
Digital Logic	**Operating System**	
Instruction Set Architecture	**Instruction Set Architecture**	
Problem Oriented Language	**Microarchitecture**	
Microarchitecture	**Digital Logic**	(lowest)

3. What **term** is used to describe each of the following concepts:

   a. The description of the **interface** between an architectural layer and the layer above it

      specification

   b. The **way** that an architectural layer performs it's work

      implementation

4. Name **two ways** that a program written at a given architectural layer can be **converted** for execution by a lower layer.

   translate (also known as compile) or interpret

5. What is the **name of the technique** that can be used to enable a machine with simple hardware to execute complex instructions?

   microcode

6. What do the acronyms **RISC** and **CISC** stand for?

   RISC – Reduced Instruction Set Computer
   CISC – Complex Instruction Set Computer

7. What are the **three** real-world computer chips used by the textbook as **sample architectures**?

   Pentium, UltraSPARC , 8051

8. Identify whether each of the computer chips listed above use **RISC** or **CISC** architectures.

   Pentium, 8051 - CISC
   UltraSPARC - RISC

9. **Which** of the following design principles are used with **RISC** systems:

   Provide lots of instructions that can perform complicated tasks

   Minimize instruction issue rate

   → Make instructions easy to decode

   → Provide lots of registers

   Make it easy for any instruction to reference memory

10. What **three major subsystems** are part of every computer?

    CPU, memory, I/O devices

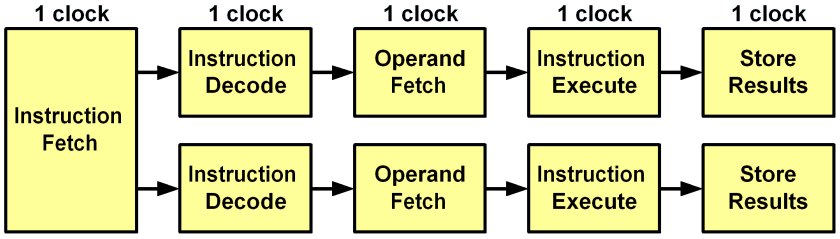
11. **How** does a CPU know which instruction to execute next?

One of the registers, usually called the Program Counter (PC) register, holds the memory address where the next instruction is located

12. **What steps** does a CPU perform in order to do the work specified by a program instruction?

fetch the instruction, execute it, and increment the PC register

13. A computer with a **2GHz clock speed** has the following pipeline:



a. What is the **bandwidth** of the pipeline?

2 instructions/clock cycle X 2 000 000 000 clock cycles/sec
= 4 000 000 000 instructions/sec (4 000 MIPS)

b. What is the **latency** of the pipeline?

each cycle takes 1 / 2 000 000 000 = 0.000 000 000 5 sec
= 0.5ns

instructions take 5 clock cycles to go thru the pipeline
5 clock cycles X 0.5ns / cycle = 2.5ns latency

14. What **problem** can occur when you build a system with many CPUs connected to a shared memory?

the memory can become a bottleneck for all of the data and (especially) instruction accesses by the CPUs