1. Are you definitely taking the course (Yes/No)?
2. Describe the von Neumann model of computing in three sentences.
   The von Neumann model of computing has four components: (1) memory (RAM), (2) ALU: arithmetic logic unit, (3) Control Unit, (4) Input/Output devices. Items (2) and (3) constitute the CPU. In this model, the program and the data are stored in the memory and instructions of the program executes sequentially.
3. What is the Amdhal's Law?
   Total Speedup = 1/(f/Speedup + (1-f)/1)
   Multicore Speedup = 1 / (f/(N * Single_Core_Performance) + (1-f)/Single_Core_Performance)
4. What is the current paradigm of general-purpose processor design?
   Multicore processor
5. What is general-purpose approximate computing?
   Definition 1: Allowing error to happen during execution so that we can run programs faster with less energy.
   Definition 2: Relaxing the abstraction of near-perfect accuracy in general-purpose computing
6. What is Dark Silicon?
   Dark silicon is the fraction of chip that needs to be powered off at all time due to power constraints.
7. What are two possible paths forward when current paradigms of general-purpose design fails?
   Approximate computing, neuromorphic computing
8. What has changed in the landscape of computing recently that motivates ACT?
   The computing industry is moving toward providing a more personalized and more targeted experience for the users.
9. What is a cache and how is it different than a register file or a scratch pad?
   A cache is an onchip memory that is hidden from the software and is not part of the architectural state and is completely managed by hardware.
   Register file and scratch pad are also on chip memory that are exposed to the software and are managed by software. Register file is also a part of the architectural state.
10. Write the recursive implementation of the Fibonacci series in any programming language.
    ```c
    int Fibonacci(int n) {
        if (n == 1) return 1;
        if (n == 2) return 1;
        return Fibonacci(n - 1) + Fibonacci(n-2); }
    ```