Preliminary Examination
Computer Architecture

Problem #1

The workload on a 2904-megahertz computer comprises three programs, A, B, and C. Program A executes 100 million instructions with a CPI of 1.20, program B executes 400 million instructions with a CPI of 1.60, and program C executes 300 million instructions at a rate of 1936 MIPS. What are the MIPS and the CPI for the total workload?

Problem #2

The instruction mix for a pipelined RISC computer is: 16% loads, 8% stores, 20% conditional branches, and 56% register arithmetic. Half of the load instructions suffer one-cycle load delays. The computer uses dynamic branch prediction with a Branch-Target-Buffer (BTB.) Assume that 90% of the branches hit the BTB and those that do hit the BTB are predicted correctly 90% of the time. A branch suffers a one-cycle delay if it doesn't hit the BTB or it is not predicted correctly by the BTB. What is the CPI of the computer if the ideal CPI is unity?

Problem #3

A computer uses a 2-way set-associative cache for instructions with a miss rate of 2.0% and a miss penalty of 20 nanoseconds. The hit time of this cache limits the clock frequency to 2 gigahertz. What is the CPI and the MIPS of this computer if the CPI without cache misses is 1.20?

Changing the I-cache of this computer to a direct-mapped cache increases the miss rate to 2.5% but reduces the hit time so the clock frequency can be raised to 2.5 gigahertz. The miss penalty is still 20 nanoseconds. With this change what is the CPI and the MIPS of the computer if the CPI without cache misses is now 1.25?

Problem #4

Floating-point operations require long pipelines in microprocessors. There are two different software techniques one can use to reduce/eliminate these data hazards when processing large data arrays:
loop-unrolling and software-pipelining.

Describe loop-unrolling and software-pipelining and point out why they are different.

As clocks get faster and faster we can expect that floating-point pipelines will get longer and longer. Which technique (loop-unrolling or software-pipelining) will be better at reducing/eliminating data hazards when the pipelines are very long?

Operating Systems

Problem #5

Describe the differences between a "process" and a "thread". Why do threads of the same process have separate stacks and program counters? Differentiate between user-level and kernel-level threads. Describe their relative advantages and disadvantages.

Problem #6

Define fairness with regard to process scheduling. Some scheduling disciplines allow unfair schedules. Consider the following disciplines:

a) First Come First Served (FCFS)
b) Round-Robin (RR)
c) Shortest Job First (SJF)
d) Shortest Remaining Time (SRT)

For each, either give an example of unfair schedule or explain why it is impossible. For a schedule example give a list of processes, their time of arrival and their burst length. If the list is infinite, state the beginning of the list and describe the pattern.

Problem #7

A certain algorithm attempts to solve the mutual exclusion (MX) problem. It selects the process with the smallest identifier that is requesting the resource and gives it exclusive access to the resource.

Is this selection method correct for solving the MX problem? If yes, explains how it satisfies the requirements of the MX problem. If not, explain which requirements are violated, give an example execution that violates the requirements and suggest modification to the algorithm that corrects the error.

Problem #8

Assume that main memory contains 4 frames and the paging system contains 10 pages. Assume that the frames are initially empty. For each of the following page replacement schemes:

a) FIFO
b) Optimal
c) Last Recently Used (LRU)

Give an example of a page reference string (a list of page references) that incurs exactly 10 page faults. Explain your answer. Note that filling an empty frame counts as a single page fault.