Architecture - A1 (25 points)

A typical workload for a given CPU includes floating point and integer instructions. Non-arithmetic operations such as jumps, loads, etc. are considered integer instructions. CPI for integer instructions is 5 and CPI for floating point instructions is 12.

a) (8 points) We are considering an enhancement to floating point instruction processing that reduces the floating point CPI to 7. This enhancement can be applied to all floating point instructions. Draw the graph depicting the relationship of the overall speedup to the percentage of the floating point instruction in the overall instruction mix. How big is the maximum achievable speedup?

b) (4 points) We are considering another enhancement that reduces the clock cycle time for all instructions by 20%. Compare this enhancement with the one from part a). When is the enhancement from part b) better than the enhancement from part a)? When is the enhancement from part a) better than the enhancement from part b)? when are they equal?

c) (8 points) We are considering another enhancement that pipelines all the instructions. CPI_ideal for integer instructions after pipelining is 2 and CPI_ideal for floating point instructions after pipelining is 4. Pipelining doubles clock cycle time. Compare this enhancement with the enhancements from part a) and b). Draw the graph depicting the relationship of the overall speedup to the percentage of the floating point instruction in the overall instruction mix. List all the cases when the enhancement from part c) is the best one.

d) (5 points) What are all the kinds of pipelining hazards that may increase instruction CPI? List each kind of a pipelining hazard, describe it in 2-3 sentences and describe in 2-3 sentences one solution that removes or alleviates the hazard.

Architecture - A2 (25 points)

Describe how the following branch prediction techniques work:

a) (3 points) Flush pipeline
b) (3 points) Always taken
c) (3 points) Always not taken
d) (3 points) Delayed branches (describe all three kinds)
e) (3 points) Local predictors
f) (3 points) Global predictors
g) (3 points) Tournament predictors
h) (4 points) What is speculative execution and how does it work in Tomasulo's algorithm? Describe this implementation in as much detail as you can.

Architecture - A3 (25 points)

You are given a system with a unified, write-allocate, two-level cache. L1 cache is write-through with a hit time of 1ns (for a word). L2 is write-back with a hit time of 10ns for a word and 15ns for L1-size block. On the average 60% of L2 blocks are dirty. We pay 100ns to transfer one L2-size block between a memory and L2 cache. On the average out of 100 accesses to the memory hierarchy, 80 of them hit in L1, and 10 of them hit in L2.

a) (2 points) What is L1 local miss rate?
b) (2 points) What is L2 local miss rate?
c) (2 points) What is L1 global miss rate?
d) (2 points) What is L2 global miss rate?
e) (5 points) what is the average memory access time for a read (instruction or data)?
f) (5 points) what is the average memory access time for a write?
g) (7 points) What would be the effect of deploying a write buffer between L1 and L2 cache that can eliminate 90% of the writes? List new values for quantities that have changed.

Architecture - A4 (25 points)

Explain how the following techniques enhance cache performance:

a) (2 points) Multilevel caches
b) (2 points) Critical word first
c) (2 points) Early restart
d) (2 points) Merging write buffer
e) (2 points) Victim caches
f) (2 points) Higher associativity
g) (2 points) Way prediction
h) (2 points) Non-blocking caches
i) (2 points) Hardware prefetching
j) (2 points) Virtual cache
k) (2 points) Trace cache
l) (3 points) What are compulsory, capacity and conflict misses, when do they occur and how can we reduce their incidence?