Indian Institute of Technology, Kharagpur
Department of Computer Science and Engineering
Mid-Semester Examination

High Performance Computer Architecture (CS60003)

Time=2 Hours **Max Marks=75**

Important Instructions:

- **Answer all questions.**
- **No clarification to any of the questions shall be provided. In case you have any queries, you can make suitable assumptions, but please write down your assumptions clearly.**
- All answers should be brief and concise. Lengthy and irrelevant answers will be penalized.

---

1. A software engineer decides to rewrite a portion of a program that accounts for 60% of execution time of the program, so that this portion can be run on multiple processors in parallel. What is the maximum speedup that the software engineer can hope to achieve? **[3]**

2. Consider a 32 bit, 5 stage MIPS processor.
   a) What is the size of cache memory required to implement a (3,2) correlating branch predictor? Assume that 4K entries have to be maintained in the prediction buffer, and one of the 4K entries would be selected based on the lower 12 bits of a branch address. **[4]**

   b) What would be the size of the cache memory required for implementing the (3,2) correlating prediction scheme of the part a) of this question, if the target address is also to be stored in the prediction table for target address prediction? **[2]**

3. Estimate the speedup that would be obtained by replacing a CPU having an average CPI (clock cycles per instruction) of 5 with another CPU having an average CPI of 3.5, with the clock period increased from 100ns to 120ns. **[5]**

4. An unpipelined processor A is a single-cycle processor (that is, CPI=1) and uses a 1GHz clock. Processor B is a pipelined version of the processor A with a 12 stage instruction pipe. Assuming ideal pipelining, what would be the clock rate of B? Give two reasons as to why B will probably not attain this clock rate. **[2+4]**

5. Identify and indicate all the *true data dependences, anti dependences , and output dependences* in the following MIPS code sequence (Mark the dependences using annotated arrows between the corresponding instructions). **[8]**

```
lw $3, 0($2)
add $3, $3, $1
lw $1, 0($2)
add $4, $3, $1
sw $4, 0($2)
```

6. A processor has a six-stage pipeline with the stages: Fetch, Decode, Register Read, Execute, Memory Access, Register Write. The processor is able to execute one instruction per cycle in the absence of branches. The branch condition and target address are both generated during the Execute pipeline stage. For a typical program, branches account for 20% of all the instructions. Of all branches, 20% are unconditional branches. Of the conditional branches, 40% are taken on the average.
   a) What will be the execution rate of this processor? Express your answer in clock cycles per instruction (CPI). **[10]**
   b) What would be the impact on CPI of using a "Not Taken static branch predictor" in the

processor? [5]

7. The table below shows the supported instruction types and the CPI of various instruction types of a 3GHz StoneBear processor. Assume that a benchmark program having a total of 2×10^9 instructions is to be run on the processor. The percentage of the different types of instructions present in this benchmark program is also shown in the table.

Instruction	Instruction Count Percentage	CPI
Integer operation	55%	1
Load/Store	30%	3
Branch	15%	4

a) What is the expected total execution time of the given benchmark program? [4]

b) A revision of the StoneBear processor raises the clock rate to 4GHz, but also at the same time increases the CPI of Load/Store instructions to 12. What speedup is expected to be achieved by this revised processor compared to the original? [6]

7. Consider the following code segment:

```
for(i=0;i<1000;i++){
        a[i]=b[i]+c[i];
        c[i]=a[i]+d[i];
}
```

a) List all the dependences (control, output, anti, and true) in the given code fragment. Indicate whether the true dependences are loop-carried or not. [3]

b) Give a scheme of how the true and false dependence can be overcome using compilation techniques to help run the for loop efficiently in a 5 stage pipelined MIPS processor. Give the restructured code. [5]

c) Can both data and control hazards be overcome for the given code segment, or at least largely reduced using compilation techniques? Briefly outline your scheme and give the restructured code. [6]

8. Identify all the data hazards in the following sequence of MIPS instructions. For each hazard, state the register involved, the writing instruction (by number) and the reading instruction (by number). Is it possible to resolve any of the hazards you identified in the previous part by reordering the instructions so that forwarding would be unnecessary? If yes, show how. If not, explain why not. [4+4]

```
add   $t0, $s0, $s1    ;1
xor   $t1, $t0, $s2    ;2
lw    $s0, -12($a0)    ;3
sub   $s5, $s0, $s1    ;4
```

---The End---