

Exam in DAT 105 (DIT 051) Computer Architecture

Time: April 4, 2018 14 – 18 in the Mechanical Engineering Building

Person in charge of the exam: Per Stenström, Phone: 0730-346340

Supporting material/tools: Chalmers approved calculator.

Exam Review: On April 18, 2018 10-12 in Room 4128

Grading intervals:

- **Fail:** Result < 24
- **Grade 3:** 24 <= Result < 36
- **Grade 4:** 36 <= Result < 48
- **Grade 5:** 48 <= Result

NOTE: Answers must be given in English

GOOD LUCK!

Per Stenström



[General disclaimer: If you feel that sufficient facts are not provided to solve a problem, either 1) ask the teacher when he visits the exam, or 2) make your own additional assumptions. Additional assumptions will be accepted if they are reasonable and required to solve the problem. Always make sure to motivate your answers.]

ASSIGNMENT 1

1A) We want to compare the performance of two computer systems A and B using the execution times of four programs P1, P2, P3 and P4. The table below lists the execution times of the programs on the two systems and a reference system R.

	P1 [s]	P2 [s]	P3 [s]	P4 [s]
System A	1	10	5	4
System B	0.5	17	2	2
System R	2	20	10	8

- i) Derive the arithmetic mean of the execution times to compare the performance of the two systems (A and B). Why is the comparison unfair? **(3 points)**
- ii) Derive the geometric mean speedup over the reference machine R for the two systems A and B. Why is this metric better than the other? **(3 points)**

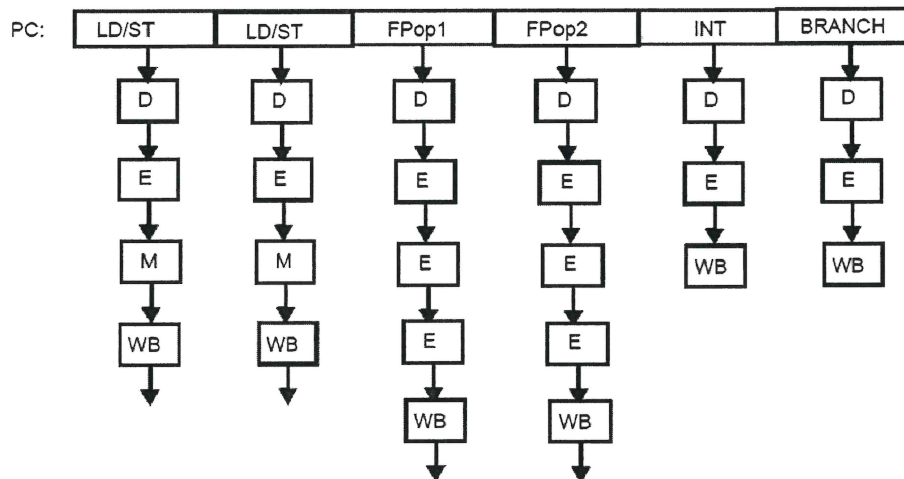
1B) A computer architect at Intel is choosing between two improvement techniques for the next generation of a microprocessor.

- **Improvement 1:** Four processors on the chip but no change to the size of the shared level 2 cache.
- **Improvement 2:** Two processors on the chip and a 50% larger shared level 2 cache.

The benchmark used to quantitatively compare the two design alternatives spend 80% of its time, when run on a single processor, in a program section that is parallelized to enjoy linear speedup on at least four processors. Moreover, measurements have shown that the CPI for all instructions, except memory instructions, is 1 and for memory instructions it is 2 on the cache assumed in improvement 1. 20% of the instructions in the benchmark are memory instructions. By making the level 2 cache 50% larger, the CPI for memory instructions goes down by 25%. Which design alternative should be chosen? **(6 points)**

ASSIGNMENT 2

We consider in this assignment a VLIW architecture that can issue two memory, two floating-point, one integer, and one branch instruction each cycle according to the pipeline organization below. There are no forwarding units.



2A) How many cycles does it take to resolve a RAW dependency between instructions of different types, i.e., between a floating-point operation and a subsequent floating point operation, between a floating-point operation and a subsequent store operation, between a load operation and a subsequent floating point operation, and between a store operation and a subsequent load operation? **(3 points)**

2B)

Consider the following simple computation:

```

LOOP: L.D   F0, 0(R1)
      ADD.D F4, F0 F2
      S.D   F4, 0(R1)
      SUBI  R1, R1, #8
      BNE   R1, R2, LOOP
  
```

We want to use software pipelining to execute the loop on the VLIW architecture. Derive the kernel of a *software-pipelined loop* by filling out the schedule table below for as many cycles needed to fill the “software pipeline” (i.e., not the VLIW pipeline). **(3 points)**

	ITE 1	ITE 2	...
INST 1			
INST 2			
...			

2C)

Under what circumstances do WAR hazards **not** occur? In case they occur, show how *Rotating Registers* can be used to avoid them by first explaining how Rotating Registers work and then by showing how the kernel is modified to make use of them. (3 points)

2D)

Consider the following code:

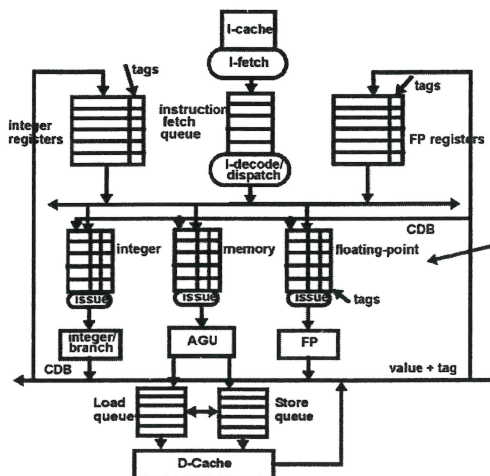
```
LW  R4, 0(R1)
ADDI R6,R4,#1
BEQ  R5,R4,LAB
LW   R6,0(R2)
LAB: SW R6, 0(R1)
```

Use *Predicated Instructions* to eliminate the branch in the code above and clearly define the semantics of the used predicated instructions. (3 points)

ASSIGNMENT 3

The diagram below shows a pipeline using Tomasulo algorithm for dynamic instruction scheduling. Assume that all integer and branch instructions execute in a single cycle whereas floating-point instructions execute in five cycles. Cache access (instruction and data) takes a single cycle whereas stores are decomposed into two instructions (one for data generation and one for cache access). Now consider the following code:

```
I1 Loop L.S F0,0(R1)
I2      L.S F1,0(R2)
I3      ADD.S F2,F1,F0
I4      S.S F2,0(R1)
I5      ADDI R1,R2,#4
I6      ADDI R2,R2,#4
I7      SUBI R3,R3,#1
I8      BNEZ R3,Loop
```



3A) How does the pipeline resolve RAW, WAR and WAW hazards? (3 points)

3B) Determine how long time it takes from the point the first instruction is dispatched until the last instruction has executed by filling out the table below: (6 points)

Instruction	Dispatch	Issue	Exec/start	Exec/compl	Cache	CDB
I1						
I2						
I3						
I4-a						
I4-b						
I5						
I6						
I7						
I8						
I9						

3C) Explain how a 2-bit branch predictor works and why it is more accurate than a 1-bit predictor for the code below by counting the number of mispredictions for the two types of predictors (3 points)

```
for(i=0; i<N; i++)
    for(j=0; j<N; j++)
```

ASSIGNMENT 4

4A) Explain the 3C classification of cache misses. For each class explain a technique that can improve the miss rate. Mention at least three different techniques. (6 points)

4B) A computer architect wants to establish the relative performance between a system with a blocking and a non-blocking cache using software prefetching. In software prefetching, prefetch instructions are appropriately inserted in the code shown below. Assume that five instructions are executed per iteration where the CPI for each instruction is one assuming that it doesn't cause a cache miss. On the other hand, if the instruction causes a cache miss,

CPI is 100. In addition, prefetch instructions result in CPI=2. Both caches have a block size of four words. In the non-blocking cache, there are 8 miss-status-holding registers.

```
for (i=0; i<1000; i++)
    C+=A[i];
```

First show how the code is annotated with prefetch instructions. How much faster does the program run on the system with a non-blocking cache using software prefetching compared to on blocking cache? A convincing explanation that is easy to follow is needed for full points. (6 points)

ASSIGNMENT 5

5A) Consider a multicore system comprising a number of processors (cores) on a chip that are connected to a single-level private cache. The private caches use the write-through write policy. $X_i=R_i$ and $X_i=W_i$ means a read and a write request to the *same* address from processor i , respectively, where $W_i=C$ means that the value C is written by processor i . Now consider the following access sequence:

$W_{1=0}$
 R_1
 R_2
 $W_{1=1}$
 R_2

What should be returned by the second read operation from processor 2 and what is the reason that the correct value is not returned given the cache write policy assumed? (3 points)

5B) How can a simple change of the write-through cache make sure that the correct value is returned to processor 2? (3 points)

5C) Multithreading refers to a general technique to switch to another thread when a high-latency operation is encountered. Explain for the three example techniques a) block multithreading b) interleaved (or fine-grained) multithreading and c) simultaneous multithreading how a five-stage pipeline must be extended to support each of the techniques. (6 points).

*** GOOD LUCK! ***