The University of Alabama in Huntsville
ECE Department
CPE 431 01
Fall 2012
Test 1 Solution

1. (1 point) A program selected for use in comparing computer performance is called a _**benchmark**_.
2. (1 point) A _**return address**_ is a link to the calling site that allows a procedure to return to the proper address.
3. (1 point) A systems program that places an object program in main memory so that it is ready to execute is called a _**loader**_.
4. (1 point) An unscheduled event that disrupts program execution is called an _**exception**_.
5. (1 point) _**Pipelining**_ is an implementation technique in which multiple instructions are overlapped in execution, much like an assembly line.

6. **(**10 points**)** Consider a computer running programs with CPU times shown in the following table.

FP Instructions	INT Instructions	L/S Instructions	Branch Instructions	Total Time
50 s	80 s	50 s	30 s	210 s

If the INT instruction time is reduced by 30 %, what is the speedup achieved?

$ET_{original}$ = 210 s
ET_{new} = 50s + 80*0.7 + 50 s + 30 s = 186 s
Speedup = $ET_{original}/ET_{new}$ = 210 s /186 s = 1.13

7. (10 points) In a von Neumann architecture, groups of bits have no intrinsic meanings by themselves. What a bit pattern represents depends entirely on how it is used. If the bit pattern 0xAF19 F329 expressed in hexadecimal notation is placed in to the Instruction Register, what MIPS instruction will be executed?

0xAF19 F329 = 1010_1111_0001_1001_1111_0011_0010_1001
Op = 101011, sw so I-type
Rs = 11000, 24 is $t8
Rt = 11001, 25 is $t9
Immediate = 1111001100101001 = $-2^{15} + 2^{14} + 2^{13} + 2^{12} + 2^{9} + 2^{8} + 2^{5} + 2^{3} + 2^{0} = -3287$

```
sw $t9, -3287($t8)
```

8. (10 points) Write down the binary representation of 120.125 in IEEE single precision format.

120.125 = 1111000.001_2
First, normalize, 1.111000001×2^6
Sign = 0, positive number
Exponent = 127 + 6 = 133, `10000101`
Fraction = `111000001`

`0100_0010_1111_0000_0100_0000_0000_0000`
0x42F0 4000

9. (10 points) In this exercise, we examine how pipelining affects the clock cycle time of the processor. Problems in this exercise assume that individual stages of the datapath have the following latencies:

IF	ID	EX	MEM	WB
200 s	170 s	220 s	210 s	150 s

What is the total latency of an `sw` instruction in a pipelined and non-pipelined processor?

Non-pipelined:
CT = IF + ID + EX + MEM + WB = 200 s + 170 s + 220 s + 210 s + 150 s = 950 s
Pipelined:
CT = 5* MAX(IF, ID, EX, MEM, WB) = 5* MAX(200, 170, 220, 210, 150)
= 5 * 220 = 1100 s

10. (10 points) The table below shows the number of instructions per processor core on a multicore processor as well as the average CPI for executing the program on 1, 2, 4, or 8 cores. Using this data, you will be exploring the speedup of applications on multicore processors.

		Cores per Processor	Instructions per Core	Average CPI
a.	1	1	1.00E+10	1.3
	2	2	5.00E+09	1.5
	4	4	2.50E+09	1.9
	8	8	1.25E+09	2.7

Assuming a 2.4 GHz clock frequency, what is the execution time of the program using 1, 2, 4, or 8 cores?

1 core:

$$ET = \frac{IC * CPI}{CR} = \frac{(1\times10^{10})1.3}{2.4\times10^9} = 5.41s$$

2 cores:

$$ET = \frac{IC * CPI}{CR} = \frac{(5\times10^{9})1.5}{2.4\times10^9} = 3.125$$

s

4 cores:

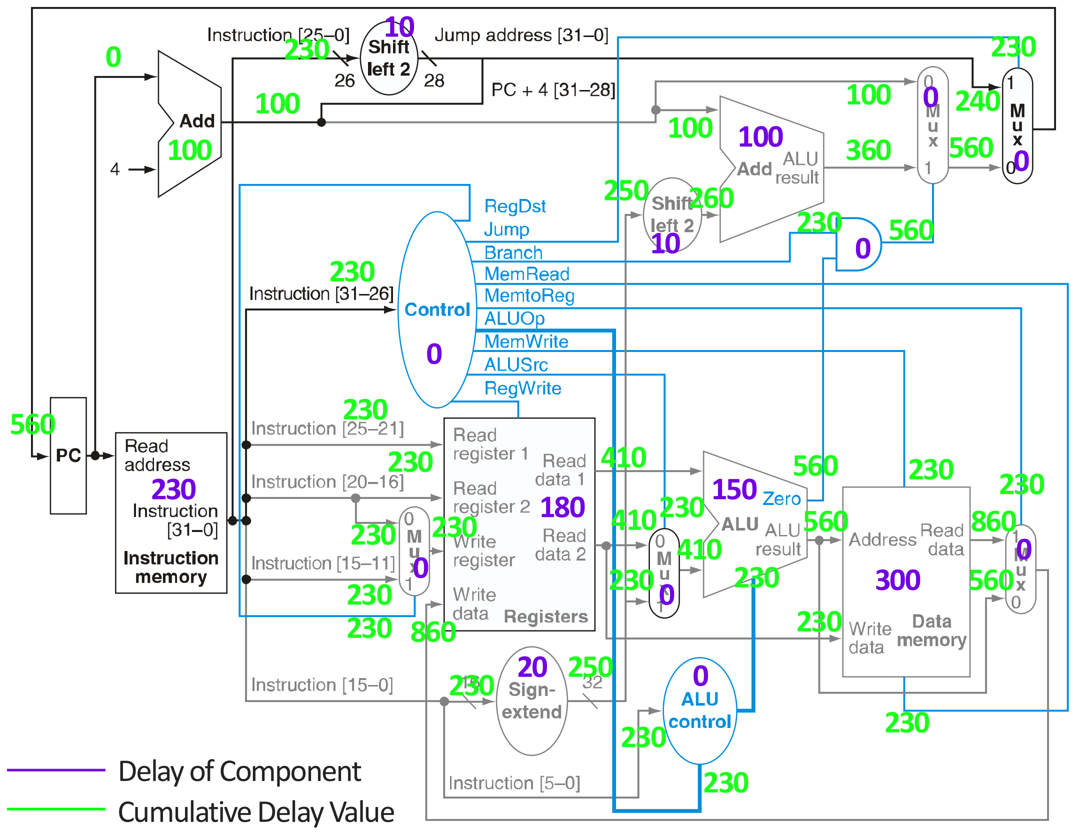
$$ET = \frac{IC * CPI}{CR} = \frac{(2.5\times10^{9})1.9}{2.4\times10^9} = 1.98s$$

8 cores:

$$ET = \frac{IC * CPI}{CR} = \frac{(1.25\times10^{9})2.7}{2.4\times10^9} = 1.41s$$

11. (15 points) In this exercise, we examine how latencies of individual components of the datapath affect the clock cycle time of the entire datapath, and how these components are utilized by instructions. For problems in this exercise, assume the following latencies for logic blocks in the datapath: Any components not listed have zero delay. Show all of your work.

I-Mem	Add	ALU	Regs	D-Mem	SignExtend	Shift-Left-2
230 ps	100 ps	150 ps	180 ps	300 ps	20 ps	10 ps



What is the clock cycle for this datapath?

Longest delay is through the ALU, data memory and back to register file. The data value is ready to be written at 860 ps, the write is complete at 1040 ps.

12. (15 points) Consider an architecture that is similar to MIPS except that it supports update addressing for data transfer instructions. If we run gcc using this architecture, some percentage of the data transfer instructions will be able to make use of the new instructions, and for each instruction changes, one arithmetic instruction can be eliminated. If 20 % of the data transfer instructions can be changed, which will be faster for gcc, the modified MIPS architecture or the unmodified architecture? Assume the CPI values shown and that the modified architecture has its cycle time increased by 15% in order to accommodate the new instructions.

Instruction Class	Average CPI	Frequency in gcc
Arithmetic	1.0	48 %
Data transfer	1.3	33 %
Conditional branch	1.7	17 %
Jump	1.1	2 %

CT_{new} = 1.15CT_{orig}
CC_{orig} = IC_{orig}(0.48 * 1.0 + 0.33 * 1.3 + 0.17 * 1.7 + 0.02*1.1)
= ICold(0.48 + 0.429 + 0.289 + 0.022) = IC_{orig} * 1.22
CC_{new} = IC_{orig}((0.48- 0.33 * 0.2) * 1.0 + 0.33 * 1.3 + 0.17 * 1.7 + 0.02 * 1.1)
= IC_{orig}(0.414 + .429 + 0.289 + 0/022) = IC_{orig} * 1.154

$$Speedup = \frac{ET_{orig}}{ET_{new}} = \frac{CC_{orig} * CT_{orig}}{CC_{new} * CT_{new}} = \frac{IC_{orig} * 1.22 * CT_{orig}}{IC_{orig} * 1.154 * 1.15 * CT_{orig}} = 0.92$$

Since the speedup is < 1, the unmodified MIPS architecture is faster.

13. (15 points) For each MIPS instruction, show the value of the opcode (op), source register (rs), and target register (rt) fields. For the I-type instructions, show the value of the immediate field, and for the R-type instructions, show the value of the destination register (rd) field.

Address	Type	op	rs	rt	rd	Immediate
`lui  $t0, 4`	I	**001111**		**01000**		**0000000000000100**
`add  $t1, $s6, $zero`	R	**000000**	**10110**	**00000**	**01001**	
`sw   $t5, 16($s7)`	I	**101011**	**10111**	**01101**		**0000000000010000**
`lw   $t2, -8($a2)`	I	**100011**	**00110**	**01010**		**1111111111111000**
`beq  $s3, $t8, -100`	I	**000100**	**10011**	**11000**		**1111111110011100**