CPSC 418	Advanced Computer Architecture	Spring 1996
	Homework 5: Due 6 Feb	

No Credit : Review Assignment

Problem 1 (0 Points) **Performance Tradeoffs and Schedules**

Your company, Boondoggle Ltd, currently offers a workstation based on your Micro1 single-issue microprocessor. The workstation is sold with a software package that includes your Comp1 C-compiler. You've just introduced the Micro2, which is a dual-issue microprocessor, and the Comp2 compiler.

1(a) (0 Points)

In the Micro2/Comp2 configuration, 50% of the instructions can dual-issue. For the next update to the Micro/Comp configuration, you are considering two options:

- Design a quad-issue machine (issue up to four instructions per cycle) named Micro3 and continue to use Comp2. Using Comp2 with Micro3, 50% of the instructions will be able to quad-issue, 10% tri-issue (3inst/cycle), 10% dual-issue, and 30% single-issue.
- Design a new compiler (Comp3) that can schedule instructions so that 80% of the instructions can dual-issue and continue to use the Micro2 design.

Which option (Micro3/Comp2 or Micro2/Comp3) is faster, how much faster is the faster option compared to the slower one and to the original (Micro2/Comp2) configuration?

1(b) (0 Points)

In order to keep up with the competition, you need to double performance every two years. The performance of the Micro2/Comp2 configuration is currently average. Boondoggle Inc only has enough money to begin to upgrade either the microprocessor (from Micro2 to Micro3) or the compiler (from Comp2 to Comp3). The hardware enhancement will take two years to develop and the software enhancement will take six months. If you begin one of the projects now, which of the two options is the better choice, in terms of its relative performance compared to the rest of the market at the end of its development?

1(c) (0 Points)

What assumptions were made in this contrived example?

Which of these assumptions were unrealistic, and why?

Problem 2 (0 Points) **MicroP Data**

For each of the following parameters:

- say whether the parameter typically increases or decreases in value as performance increases (*e.g.* is 'bigger is better"?)
- say which one of the values given is typical for a workstation class computer today

Parameter	Big=Better?				
SPECint92		30	100	300	1000
Fabrication feature size		$200mm^2$	$100mm^2$	$0.01\mu m$	$0.35\mu m$
# Pipe stages		3	10	20	30
# Execution units		1	2	5	10
Clock speed		10MHz	20MHz	40MHz	100MHz
L1 Cache		16Kb	64Kb	128Kb	1M
L2 Cache		16Kb	128Kb	256Kb	2M

Problem 3 (0 Points) **Addressing Modes**

Describe the best use of, and give an example of (*e.g.* sample assembly language line with descriptive comment), each of the three different addressing modes (absolute, displacement, indexed) for both data transfers (loads/stores) and control (jumps).

Problem 4 (0 Points) **Function Calls**

Given the following functions in pseudo-code:

```
int g(int M) {
    return (2*M)
}

void f(int *X, int Y) {
    X = g( Y )
}

void main () {
   ...
   A = 3+ B
   f(&A, B-2)
   A = A + B ...
}
```

Write the assembly code for a computer with eight registers (R0 – R8) and the following definitions/conventions:

param/scratch regs	R0, R1
return address (RA)	R5
stack pointer (SP)	R6
program counter (PC)	R7

The addresses of `A` and `B` are initially in `R2` and `R3` respectively.

Problem 5 (0 Points) **Traps / Interupts**

You have just finished writing a new keyboard device driver and are testing it using a simulator.

- The interupt vector table begins at address 1184 and has 4k entries.
- The interupt vector for the keyboard driver is 2032.
- The keyboard driver subroutine trace is 500 instructions long.

Memory

Addr	Contents
⋮	⋮
1184	20310
1185	887344
⋮	⋮
2031	580400
2032	581300
2033	579000
⋮	⋮
3214	890060
3215	890560
3216	891254
⋮	⋮

Describe a test you could run to check if interupts are working properly and the keyboard driver is able to read a character from the keyboard. List what you would examine (*e.g.* program counter, register contents, etc), the values you expect to see, and the order or approximate time you would expect to see them. Remember, you are running on a simulator of the CPU, so you can examine *any* hardware you want.

Problem 6 (0 Points) **Communicating Processes**

Briefly, but at a low level, (*e.g.* O/S and hardware) describe a simple, fast, and lightweight mechanism (*i.e. not* named pipes) that would allow one process to send sequences of commands and data to another process.

What instructions, registers, *etc.* should the CPU provide to facilitate the above mechanism?

Problem 7 (0 Points) **Disk / Files**

As companies move towards "groupware", where several people can work on the same document at the same time, text editors are offering facilities to "lock" files for users so that two people don't try to modify the same file at the same time. Many ISAs offer semaphore instructions (*e.g.* LODSET in F&R).

Is LODSET the best way to "lock" a file? Justify your answer.