

## Computer Organization and Structure

Homework #2  
Due: 2003/11/4

1. We wish to compare the performance of two different machines: M1 and M2. The following measurements have been made on these machines:

Program	Time on M1	Time on M2
1	10 seconds	5 seconds
2	3 seconds	4 seconds

Which machine is faster for each program and by how much? Consider the two machines and programs as the above. The following additional measurements were made:

Program	Instructions executed on M1	Instructions executed on M2
1	$200 \times 10^6$	$160 \times 10^6$

Find the instruction execution rate (instructions per second) for each machine when running program 1. If the clock rates of machines M1 and M2 are 200MHz and 300MHz, respectively, find the clock cycles per instruction (CPI) for program 1 on both machines using the data as the above.

2. Consider two different implementations, M1 and M2, of the same instruction set. There are four classes of instructions (A, B, C, and D) in the instruction set. M1 has a clock rate of 500 MHz. The average number of cycles for each instruction class on 1 is as follows:

Class	CPI for this class
A	1
B	2
C	3
D	4

M2 has a clock rate of 750 MHz. The average number of cycles for each instruction class on 1 is as follows:

Class	CPI for this class
A	2
B	2
C	4
D	4

Assume that peak performance is defined as the fastest rate that a machine can execute an instruction sequence chosen to maximize that rate. What are the peak performances of M1 and M2 expressed as instructions per second?

3. We are interested in two implementations of a machine one with and one without special

floating-point hardware. Consider a program, P, with the following mix of operations:

floating-point multiply	10%
floating-point add	15%
floating-point divide	5%
integer instructions	70%

Machine MFP (Machine with Floating Point) has floating-point hardware and can therefore implement the floating-point operations directly. It requires the following number of clock cycles for each instruction class:

floating-point multiply	6
floating-point add	4
floating-point divide	20
integer instructions	2

Machine MNFP (Machine with No Floating Point) has no floating-point hardware and so must emulate the floating-point operations using integer instructions. The integer instructions all take 2 clock cycles. The number of integer instructions needed to implement each of the floating-point operations is as follows:

floating-point multiply	30
floating-point add	20
floating-point divide	50

Both machines have a clock rate of 1000 MHz. Find the native MIPS ratings for both machines.

4. The table below shows the number of floating-point operations executed in two different programs and the runtime for those programs on three different machines:

Program	Floating-point operations	Execution time in seconds		
		Computer A	Computer B	Computer C
Program 1	10,000,000	1	10	20
Program 2	100,000,000	1000	100	20

Which machine is fastest according to total execution time? How much faster is it than the other two machines?

5. The following code fragment processes an array and produces two important values in registers \$v0 and \$v1. Assume that the array consists of 5000 words indexed 0 through 4999, and its base address is stored in \$a0 and its size (5000) in \$a1. Describe in one sentence what this code does. Specifically, what will be returned in \$v0 and \$v1?

```

add    $a1, $a1, $a1
add    $a1, $a1, $a1
add    $v0, $zero, $zero

```

```

outer:    add    $t0, $zero, $zero
         add    $t4, $a0, $t0
         lw     $t4, 0($t4)
         add    $t5, $zero, $zero
         add    $t1, $zero, $zero
inner:    add    $t3, $a0, $t1
         lw     $t3, 0($t3)
         bne   $t3, $t4, skip
         addi   $t5, $t5, 1
skip:     addi   $t1, $t1, 4
         bne   $t1, $a1, inner
         slt   $t2, $t5, $v0
         bne   $t2, $zero, next
         add   $v0, $t5, $zero
         add   $v1, $t4, $zero
next:     addi   $t0, $t0, 4
         bne   $t0, $a1, outer

```

Assume that the above code is run on a machine with a 500 MHz clock that requires the following number of cycles for each instruction:

Instruction	Cycles
add, addi, slt	1
lw, bne	2

In the worst case, how many seconds will it take to execute this code?

6. Show the single MIPS instruction or minimal sequence of instructions for this C statement:

$$x[10] = x[11] + c;$$

Assume that  $c$  corresponds to register  $\$t0$  and the array  $x$  has a base address of  $4,000,000_{\text{ten}}$ .

7. Consider the following fragment of C code:

```
for (i = 0; i <= 100; i = i + 1) { a[i] = b[i] + c; }
```

Assume that  $a$  and  $b$  are arrays of words and the base address of  $a$  is in  $\$a0$  and the base address of  $b$  is in  $\$a1$ . Register  $\$t0$  is associated with variable  $i$  and register  $\$s0$  with  $c$ . Write the code for MIPS. How many instructions are executed during the running of this code? How many memory data references will be made during execution?

8. Write a procedure, `bfind`, in MIPS assembly language. The procedure should take a single argument that is a pointer to a null-terminated string in register  $\$a0$ . The `bfind` procedure should locate the first `b` character in the string and return its address in register  $\$v0$ . If

there are no b's in the string, then bfind should return a pointer to the null character at the end of the string. For example, if the argument to bfind points to the string "imbibe," then the return value will be a pointer to the third character of the string.