**Assignment# 1**

**CSE-332/Summer 2020 (Due date: Friday, July 31, 6pm)**

1. Consider two different implementations of the same instruction set architecture. There are four classes of instructions, A, B, C, and D. The clock rate and CPI of each implementation are given in the following table.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | CLOCK RATE | CPI class A | CPI class B | CPI class C | CPI class D |
| P1 | 2.5 GHz | 1 | 2 | 3 | 3 |
| P2 | 3 GHz | 2 | 2 | 2 | 2 |

1. Given a program with 106 instructions divided into classes as follows: 10% class A, 20% class B, 50% class C, and 20% class D, which implementation is faster?
2. What is the global CPI for each implementation?
3. Find the clock cycles required in both cases.

Solution:

Class A: 10%

Class B: 20%

Class C: 50%

Class D: 20%

For P1: CPI = 0.1× 1+ 0.2× 2 + 0.5 × 3 + 0.2 × 3 = 2.6

Execution Time: 106 × 2.6 × (1/2.5 × 109) second = 275.6 × 0.4 × 10-9 sec = 110.24 × 10-9 sec

Clock cycles: 2.6 × 106 = 275.6

For P2: CPI = 0.1× 2+ 0.2× 2 + 0.5 × 2 + 0.2 × 2 = 2.0

Execution Time: 106 × 2.0 × (1/3 × 109) second = 212 × 0.33 × 10-9 sec = 69.96 × 10-9 sec

Clock cycles: 2.0× 106 = 212

1. P2 implementation is faster
2. For P1: Global CPI = 2.6

For P2: Global CPI = 2.0

1. For P1: Clock cycles: 275.6

For P2: Clock cycles: 212

1. The following table shows the number of instructions for a program.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | ARITH | STORE | LOAD | BRANCH | TOTAL |
| A. | 650 | 100 | 600 | 50 | 1400 |
| B. | 750 | 250 | 500 | 500 | 2000 |

1. Assuming that arith instructions take 1 cycle, load and store 5 cycles, and branches 2 cycles, what is the execution time of the program in a 2 GHz processor?
2. Find the CPI for the program.
3. If the number of load instructions can be reduced by one half, what is the speedup and the CPI?

Solution:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | ARITH | STORE | LOAD | BRANCH | TOTAL |
| A. | 650 | 100 | 600 | 50 | 1400 |
| B. | 750 | 250 | 500 | 500 | 2000 |
|  | CPI=1 | CPI=5 | CPI=5 | CPI=2 |  |

For A:

Average CPI for A: (650 × 1 + 100 × 5 + 600 × 5 + 50 × 2)/1400 = 4200/1400 = 3

Execution time for A: 1400 × 3 × 1/(2 × 109) = 4200 × 0.5 × 10-9 sec = 2100 ns

If the number of load instructions can be reduced by one half,

Average CPI for A: (650 × 1 + 100 × 5 + 300 × 5 + 50 × 2)/1400 = 2700/1100 = 2.45

Speed up = (1400 × 3 × 1/(2 × 109))/ (1100 × 2.45 × 1/(2 × 109)) = 4200/2695 = 1.55

For B:

Average CPI for B: (750 × 1 + 250 × 5 + 500 × 5 + 500 × 2)/2000 = 5500/2000 = 2.75

Execution time for A: 2000 × 2.75 × 1/(2 × 109) = 2750 ns

If the number of load instructions can be reduced by one half,

Average CPI for B: (750 × 1 + 250 × 5 + 250 × 5 + 500 × 2)/1750 = 4250/1750 = 2.42

Speed up = (2000 × 2.75× 1/(2 × 109))/ (1750 × 2.42 × 1/(2 × 109)) = 5500/4235 = 1.30

1. A benchmark program is run on a 40 MHz processor. The executed program consists of 100,000 instruction executions, with the following instruction mix and clock cycle count:

|  |  |  |
| --- | --- | --- |
| INSTRUCTION TYPE | INSTRUCTION COUNT  IN MILLIONS | CYCLES PER INSTRUCTIONS |
| ARITHMETIC | 45000 | 1 |
| DATA TRANSFER | 32000 | 2 |
| FLOATING POINT | 15000 | 2 |
| CONTROL TRANSFER | 8000 | 2 |

Determine the effective CPI, MIPS rate, and execution time for this program.

Solution:

CPI = (45000 × 1 + 32000 × 2 + 15000 × 2 + 8000 × 2)/100000 = 1.55

Execution time = 100000 × 1.55 × (1/40 × 106) = 0.003875 sec = 3.875 ms

MIPS = (100000 × 40 × 106)/(100000 × 1.55 × 106) = 25.80

1. Consider two different machines, with two different instruction sets, both of which have a clock rate of 200 MHz. The following measurements are recorded on the two machines running a given set of benchmark programs:

|  |  |  |  |
| --- | --- | --- | --- |
| MACHINE - A | INSTRUCTION TYPE | INSTRUCTION COUNT | CYCLES PER INSTRUCTIONS |
| ARITHMETIC | 8 | 1 |
| LOAD and STORE | 4 | 3 |
| BRANCH | 2 | 4 |
| OTHERS | 4 | 3 |

...............

|  |  |  |  |
| --- | --- | --- | --- |
| MACHINE - B | INSTRUCTION TYPE | INSTRUCTION COUNT | CYCLES PER INSTRUCTIONS |
| ARITHMETIC | 10 | 1 |
| LOAD and STORE | 8 | 2 |
| BRANCH | 2 | 4 |
| OTHERS | 4 | 3 |

Determine the effective CPI, MIPS rate, and execution time for each machine.

Machine-A

Effective CPI: (8×1+4×3+2×4+4×3)/18 = 40/18 = 2.22

Execution time: 18 × 2.22 × (1/200 × 106) = 0.198 × 10-6 sec

MIPS = f /(CPI × 106) = (200 × 106)/(2.22 × 106) = 90.09

Machine-B

Effective CPI: (10×1+8×2+2×4+4×3)/18 = 46/24 = 1.92

Execution time: 24 × 1.92 × (1/200 × 106) = 0.23 × 10-6 sec

MIPS = f /(CPI × 106) = (200 × 106)/(1.92 × 106) = 104.16

1. Assuming the following assembly instructions are executed in the order that they are found in the table below, fill out the chart indicating the stage of the standard 5-stage pipeline that the instruction will be in during the clock cycles. If an instruction is not in any stage during a cycle, simply leave that box blank. Indicate the stages of the pipeline using: IF, ID, MEM, EX, and WR. If there is gap in stages due to a data hazard, make sure to indicate the data hazard using d\*.

Solution:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Instruction\ cycle** | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| ADD R10, R2, R3 | IF | ID | MEM | EX | WR |  |  |  |  |
| SUB R4, R1, R5 |  | IF | ID | MEM | EX | WR |  |  |  |
| AND R6, R4, R7 |  |  | IF | ID | MEM | d\* | EX | WR |  |
| OR R8, R12, R9 |  |  |  | IF | ID |  | MEM | EX | WR |
| XOR R10, R1, R11 |  |  |  |  | IF |  | ID | MEM | EX |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

Note: Sum of R1 and R5 must be stored in R4 first (instruction: SUB R4, R1, R5) and then the updated value of R4 can be used in following instruction (instruction: AND R6, R4, R7). So there will be a data hazard marked by d\*. As a result, following instructions will be delayed by one time step.

1. Assume a machine using Five-stage pipelining runs a program. Instruction-3 is a conditional branch instruction. If the condition is TRUE, CPU skips next three instructions. Instruction-8 is also a conditional branch instruction and if it is FALSE, program control returns to Instruction-4. Show the time steps of pipelining stages assuming that both Instructions 3 and 8 are evaluated evaluated TRUE.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Instructions** | **Time Steps** | | | | | | | | | | | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** |
| Inst-1 | IF | ID | MEM | EX | WR |  |  |  |  |  |  |  |  |  |  |  |  |
| Inst-2 |  | IF | ID | MEM | EX | WR |  |  |  |  |  |  |  |  |  |  |  |
| Inst-3 |  |  | IF | ID | MEM | EX | WR |  |  |  |  |  |  |  |  |  |  |
| Inst-4 |  |  |  | IF | ID | MEM | cleared | | |  |  |  |  |  |  |  |  |
| Inst-5 |  |  |  |  | IF | ID |  |  |  |  |  |  |  |  |
| Inst-6 |  |  |  |  |  | IF |  |  |  |  |  |  |  |  |
| Inst-7 |  |  |  |  |  |  | IF | ID | MEM | EX | WR |  |  |  |  |  |  |
| Inst-8 |  |  |  |  |  |  |  | IF | ID | MEM | EX | WR |  |  |  |  |  |
| Inst-9 |  |  |  |  |  |  |  |  | IF | ID | MEM | EX | WR |  |  |  |  |
| Inst-10 |  |  |  |  |  |  |  |  |  | IF | ID | MEM | EX |  |  |  |  |
| Inst-11 |  |  |  |  |  |  |  |  |  |  | IF | ID | MEM |  |  |  |  |  |
| Inst-12 |  |  |  |  |  |  |  |  |  |  |  | IF | ID |  |  |  |  |
| Inst-13 |  |  |  |  |  |  |  |  |  |  |  |  | IF |  |  |  |  |

|  |
| --- |
|  |

1. Consider two different machines, with two different instruction sets, using two compilers runs a program having following distribution of instructions.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Instruction Type | Machine-A (Clock rate 250MHz) | | Machine-B (Clock rate 300MHz) | |
| Instruction Count in Millions | CPI | Instruction Count in Millions | CPI |
| ARITHMETIC | 8 | 1 | 7 | 1 |
| LOAD | 6 | 5 | 8 | 4 |
| BRANCH | 2 | 4 | 2 | 5 |
| STORE | 4 | 2 | 3 | 3 |

Determine the effective CPI, MIPS rate, and execution time for each machine. Do MIPS ratings reflect the true relative performances of Machines A & B, justify your answer with reference to other values and performance measures.

Solution:

Machine-A

Effective CPI: (8×1+6×5+2×4+4×2)/20 = 54/20 = 2.7

Execution time: 20 ×106× 2.7 × (1/250 × 106) = 0.216 sec

MIPS = f /(CPI × 106) = (250 × 106)/(2.7 × 106) = 92.59

Machine-B

Effective CPI: (7×1+8×4+2×5+3×3)/20 = 58/20 = 2.9

Execution time: 20 ×106× 2.9 × (1/300 × 106) = 0.193 sec

MIPS = f /(CPI × 106) = (300 × 106)/(2.9 × 106) = 103.44

Here, MIPS ratings reflect the true relative performances of Machines A & B. Machine-B is faster since execution time is lower.

1. A computer *M*2 has the following CPIs for instruction types A thru D, and a program *P*3 has the following mix of instructions.

*M*2: Type A CPIA = 1.7 Type B CPIB = 2.1 Type C CPIC = 2.7 Type D CPID = 2.4

*P*3: Type A = 22% Type B = 29% Type C = 17% Type D = remaining %

1. Calculate the average CPI of Machine M2
2. Calculate the runtime of *P*3 on *M*2 if IC = 22,311 and clock rate is 3.3 GHz

Solution:

1. Average CPI: 0.22 × 1.7 + 0.29 × 2.1 + 0.17×2.7 + 0.32 × 2.4 = 0.374+0.609+ 0.459+0.768 = 2.21
2. Run time: 22311 × 2.21 × (1/3.3 × 109) = 14.94 ×10-6 sec
3. You are on the design team for a new processor. The clock of the processor runs at 200 MHz. The following table gives instruction frequencies for Benchmark program, as well as how many cycles the instructions take, for the different classes of instructions. For this problem, we assume that (unlike many of today's computers) the processor only executes one instruction at a time.

|  |  |  |
| --- | --- | --- |
| Instruction Type | Frequency | CPI |
| Loads & Stores | 30% | 6 |
| Arithmetic Instructions | 50% | 4 |
| All Others | 20% | 3 |

1. Calculate the CPI for Benchmark program.
2. What is the MIPS rating of the processor speed?
3. The hardware expert says that if you double the number of registers, the cycle time must be increased by 20%. What would the new clock speed be (in MHz)?
4. The compiler expert says that if you double the number of registers, then the compiler will generate code that requires only half the number of Loads & Stores. What would the new CPI be on the benchmark?
5. How many CPU seconds will the benchmark take if we double the number of registers (taking into account both changes described above)?

Solution:

1. CPI: 0.3 × 6 +0.5×4+0.2×3 = 4.4
2. MIPS = (200 × 106)/(4.4 × 106) = 45.45
3. Old Cycle time : 1/(200 × 106) = 0.005 × 10-6sec

New Cycle time: 0.005 × 10-6sec + 0.001 × 10-6sec = 0.006 × 10-6sec

New clock speed, f = 1/(0.006 × 10-6sec) Hz = 166.66 × 106 Hz = 166.66 MHz

1. Let us assume that, initially total number of instructions was = 100 (Loads & Stores = 30, Arithmetic = 50 and Others = 20).

Half the number of Loads & Stores = 30/2 = 15

New CPI: (15 × 6 + 50×4 + 20×3)/85 = (90+200+60)/85 = 350/85 = 4.11

1. Taking c (New clock speed = 166.66 MHz and d (New CPI = 4.11) into consideration,

Execution time: 85 × 4.11 × 1/(166.66 × 106) sec = 2.10 × 10-6 sec