Computer Architecture Homework 2
Student: Chao Chen

1. The techniques used in contemporary processors for performance optimizing includes:

1). Pipelining:
Instruction is split up into a sequence of steps so different steps can be executed in parallel and instructions can be processed concurrently.

2). Branch prediction
Processor looks ahead and tries to predict which code branch to go for next step before it's known for sure. It plays a critical role in achieving high effective performance in many modern pipelined microprocessor architectures such as x86.

3). Superscalar execution
A superscalar processor executes more than one instruction during a clock cycle by simultaneously dispatching multiple instructions to different functional units on the processor.

4). Data flow analysis
Processor analyzes which instructions are dependent on each other's results, or data, to create an optimized schedule of instructions .

5). Speculative execution
Speculative execution is an optimization technique where a computer system performs some task that may not be actually needed. The main idea is to do work before it is known whether that work will be needed at all, so as to prevent a delay that would have to be incurred by doing the work after it is known whether it is needed. If it turns out the work was not needed after all, any changes made by the work are reverted and the results are ignored.

2. Amdahl's Law is a law governing the speedup of using parallel processors on a problem, versus using only one serial processor:

$$S = 1 / ((1 - f) + f / N)$$

where S is the theoretical speedup in latency of the execution of the whole task, f is the percentage of the parallelizable part of the program in time scale and N is the number of processors. It shows that the theoretical speedup of the execution of the whole task increases with the improvement of the resources of the system and that regardless the magnitude of the improvement, the theoretical speedup is always limited by the part of the task that cannot benefit from the improvement.

3. Little's Law:

$$L = \lambda W$$

Average number (in long-term) of items in a queuing system L, equals the average rate at which items arrive λ multiplied by the time that an item spends in the system w.

4.

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

MIPS = f / CPI x 10^6 = 40 x 10^6 / 1.55 x 10^6 = 25.81

T = CPI x Ic x (1/f) = 1.55 * 100000 * (1 / 40 x 10^6) = 0.003875

b) Machine A:

CPI = (8x1 + 4x3 + 2x4 + 4x3) x 10^6 / 18 x 10^6 = 2.22

MIPS = f / CPI x 10^6 = 200 / 2.22 = 90.09

T = CPI x Ic x (1/f) = 40 x 10^6 x (1 / 200 x 10^6) = 0.2 sec

Machine B:

CPI = (10x1 + 8x2 + 2x4 + 4x3) x 10^6 / 24 x 10^6 = 1.92

MIPS = f/CPI x 10^6 = 200 / 1.92 = 104.17

T = CPI x Ic x (1/f) = 46 x 10^6 x (1 / 200 x 10^6) = 0.23 sec

Comment:

Machine B is faster in instruction execution than Machine A because of the smaller CPI (a single instruction takes less clock cycles), but the execution time of the benchmark of Machine B is longer since the total instructions of the program running on Machine B is more than on Machine A.

c). a. speed metric:

	R	M	Z
E	1.0	0.59	0.32
F	1.0	0.84	0.84
H	1.0	2.32	2.05
I	1.0	0.90	1.67
K	1.0	0.48	0.48
A Mean	1.0	1.03	1.07

b.

	R	M	Z
E	1.71	1.0	0.55
F	1.19	1.0	1.0
H	0.43	1.0	0.88
I	1.11	1.0	1.86
K	2.10	1.0	1.00
A Mean	1.31	1.0	1.06

c. On reference system a(R), Machine Z is the slowest; On reference system b(M) Machine R is the slowest.

d.

	R	M	Z
E	1.0	0.59	0.32
F	1.0	0.84	0.84
H	1.0	2.32	2.05
I	1.0	0.90	1.67
K	1.0	0.48	0.48
G Mean	1.0	0.87	0.85

	R	M	Z
E	1.71	1.0	0.55
F	1.19	1.0	1.0
H	0.43	1.0	0.88
I	1.11	1.0	1.86
K	2.10	1.0	1.00
G Mean	1.15	1.0	0.98

In these two calculations based on geometric mean, for the reference system (R), Machine R is the slowest, for the reference system (M), Machine R is the slowest.

d). According to Little's Law: $L = \lambda W$

$8 = 18 \times W$

thus, $W = 0.44$ hours = 26 minutes

Each customer in average spends 26 mins in the shop.