Michael Zhou
UID: 804 663 317
1/17/2018

CSM151B Homework 1
1.5, 1.6, 1.7, 1.13

1.5

a)

Processor:	Clock Rate:	Cycle Per Instruction:
P1	3.0GHz	1.5
P2	2.5GHz	1.0
P3	4.0GHz	2.2

We have: Performance = Clock Rate/Cycles Per Instruction

Use the above formula to calculate the performance of each processor:

Performance (P1) = 3.0*10^9 / 1.5 = 2.0*10^9 Instruction Per Second
Performance (P2) = 2.5*10^9 / 1.0 = 2.5*10^9 Instruction Per Second
Performance (P3) = 4.0*10^9 / 2.2 = 1.81*10^9 Instruction Per Second

Therefore, Processor2 results in the highest performance in terms of instructions per second.

b)
Running a program for 10 seconds:

Number of Cycles = Clock Rate * Time (s)

Number of Cycles (P1) = 3.0*10^9*10 = 3.0*10^10 Cycles
Number of Cycles (P2) = 2.5*10^9*10 = 2.5*10^10 Cycles
Number of Cycles (P3) = 4.0*10^9*10 = 4.0*10^10 Cycles

Number of instructions = Number of Cycles / Cycles Per Instruction

Number of instructions (P1) = 3.0*10^10 / 1.5 = 2.0*10^10
Number of instructions (P2) = 2.5*10^10 / 1.0 = 2.5*10^10
Number of instructions (P3) = 4.0*10^10 / 2.2 = 1.81*10^10

c)

To reduce the execution time by 30% but this leads to an increase of 20% in the CPI.

ET = IC * CPI * CT = (IC * CPI) / Clock Rate
Clock Rate = IC *CPI / ET

For Processor 1:

NEW CPI = 1.2* CPI
NEW ET = 0.7 ET
Therefore:
NEW Clock Rate = 1.2/0.7 IC*CPI/ET = 1.714* Clock Rate = 1.714*3.0*10^9 = 5.14GHz
The new Clock Rate required for P1 to get the time reduction is 5.14GHz.

For Processor 2:
NEW CPI = 1.2* CPI
NEW ET = 0.7 ET
Therefore:
NEW Clock Rate = 1.2/0.7 IC*CPI/ET = 1.714* Clock Rate = 1.714*2.5*10^9 = 4.28GHz
The new Clock Rate required for P1 to get the time reduction is 4.28GHz.

For Processor 3:
NEW CPI = 1.2* CPI
NEW ET = 0.7 ET
Therefore:
NEW Clock Rate = 1.2/0.7 IC*CPI/ET = 1.714* Clock Rate = 1.714*4.0*10^9 = 6.85GHz
The new Clock Rate required for P1 to get the time reduction is 6.85GHz.

1.6

Processor	Clock Rate	Class A	Class B	Class C	Class D
P1	2.5GHz	1	2	3	3
P2	3GHz	2	2	2	2

10% class A, 20% class B, 50% class C, and 20% class D

Weighted Average CPI for Processor 1:
CPI(avg.) = 1*0.1 + 2*0.2 + 3*0.5 + 3*0.2 = 2.6

Weighted Average CPI for Processor 2:
CPI(avg.) = 2*0.1 + 2*0.2 + 2*0.5 + 2*0.2 = 2

ET = IC * CPI * CT = (IC * CPI) / Clock Rate
Calculated the estimated CPU time:

ET(P1) = 1.0*10^6*2.6/(2.5*10^9) = 1.04 ms

ET(P2) = 1.0*10^6*2/(3*10^9) = 0.667 ms

Thus we see that processor 2 runs faster.

a)

Global CPI = CPI time*Clock Rate/(Numbers of Instructions)

Global CPI for Processor 1 = 1.04*10^(-3)*2.5*10^9/10^6 = 2.6
The global CPI for P1 is 2.6.

Global CPI for Processor 2 = 0.667*10^(-3)*3*10^9/10^6 = 2.0
The global CPI for P2 is 2.0.

b)
Number of Clock Cycles required:
Clock Cycles for P1 = Global CPI * No. of Instructions = 2.6*10^6 Cycles

Clock Cycles for P2 = Global CPI * No. of Instructions = 2.0*10^6 Cycles

1.7
a)

Compiler	Execution Time	Instruction Count
A	1.1s	1*10^9
B	1.5s	1.2*10^9

Cycle Time = 1ns = 1.0*10^(-9) s

ET = IC * CPI * CT
Thus, CPI = ET/ (IC * CT)

CPI for Compiler A:
CPI = ET/ (IC * CT) = 1.1/ (1*10^9*1.0*10^(-9)) = 1.1

CPI for Compiler B:
CPI = ET/ (IC * CT) = 1.5/ (1.2*10^9*1.0*10^(-9)) = 1.25

b)

ET (A) = IC * CPI * CT (A)
ET (B) = IC * CPI * CT (B)
Because the execution time is the same, equate both equations
IC(A) * CPI(A) * CT (A) = IC(B) * CPI(B) * CT (B)

CT(A)/CT(B) = IC(B) * CPI(B)/ (IC(A) * CPI(A)) = 1.2*10^9*1.25/(1*10^9*1.1) = 1.3636 Times

Therefore, the Clock Cycle of Compiler A is 1.3636 times faster than Clock Cycle of the Compiler B.

c)
The new complier has
IC = 0.6*10^(9)
Avg. CPI = 1.1
Cycle Time = 1ns = 1.0*10^(-9) s

ET (NEW) = IC * CPI * CT = 0.6*10^(9)*1.1*1.0*10^(-9) = 0.66s

Comparing to Compiler A:
ET(NEW)/ ET(A) = 1.1s/0.66s = 1.67
Therefore, the Execution Time of the new Compiler is 1.67 times faster than the Execution Time of the Compiler A.

Comparing to Compiler B:
ET(NEW)/ ET(B) = 1.5s/0.66s = 2.27
Therefore, the Execution Time of the new Compiler is 2.27 times faster than the Execution Time of the Compiler B.

1.13

1.13.1)
% Total Time Reduced = Time Reduced in FP Operation/ Total Time
=(20%)*70/ 250= 0.056 = 5.6%

1.13.2)
Reduced time = 250s * 20% = 50s
Original INT time = 250s - 70s - 85s - 40s = 55s
INT operation time reduced by = 50s/55s = 90.0%

1.13.3)
Total time reduced by 20% = 0.8*250 = 200s
Reducing The time of branch instruction = 85 + 70 + 55 = 210s

Since 210 > 200, an contradiction occurs. Therefore, it is not possible to reduce the total time by 20% by simply reducing the time of branch instruction.