EECS 112 & CSE 132, FALL 2016
HW1: Chapter 1

Student ID:									Name:

Notes:

- Questions are designed part by part to help you through the problem
- Anything outside the boxes will not be graded.
- Empty boxes without an asterisk (*) are to show your work briefly.
- Empty boxes with an asterisk (*) are to show your final result only.
- Students who turn in neat papers in good handwriting tend to get higher grades. The opposite holds as well!

1- A manufacturer can choose between two wafer types W1 and W2, both of radius 30cm.

Type	Probability of at least one defect per square cm	Price per square cm
W1	.001	$20
W2	.01	$1

The manufacturer also can choose between two fabrication technologies T1 and T2:

technology	Die size in square cm	Processing cost per die	Processing defect rate	Bonding cost per die	Bonding defect rate
T1	1	$50	.01	$15	.01
T2	3	$30	.005	$10	.008

1a) What is the wafer cost for each wafer choice?

W1	Pi*(30^2)*20	*$56,549
W2	Pi*(30^2)*1	*$2,827

1b) How many dies fit on a wafer in each technology?

T1	Pi*(30^2)/1	*2827
T2	Pi*(30^2)/3	*942

1c) What is the total processing cost in each technology?

T1	2827*50	*$141,350
T2	942*30	*$28.260

1d) What fraction of dies survive after processing, for each wafer/technology combination?

W1,T1	1-[.001*1+.01-.001*1*.01]	*.9890
W1,T2	1-[.001*3+.005-.001*3*.005]	*.9920
W2,T1	1-[.01*1+.01-.01*1*.01]	*.9801
W2,T2	1-[.01*3+.005-.01*3*.005]	*.9651

1e) what is the total cost (wafer + processing) per working die before bonding for each wafer/Technology combination?

W1,T1	($141,350+$56,549)/(.9890*2827)	*$70.78
W1,T2	($28.260+$56,549)/(.9920*942)	*$90.76
W2,T1	($141,350+$2,827)/(.9801*2827)	*$52.04
W2,T2	($28.260+$2,827)/(.9651*942)	*$34.19

1f) what is the total cost per working packaged die for each combination?

W1,T1	(70.78+15)/(1-.01)	*$86.65
W1,T2	(90.76+10)/(1-.008)	*$101.57
W2,T1	(52.04+15)/(1-.01)	*$67.71
W2,T2	(34.19+10)/(1-.008)	*$44.55

1g) Which wafer/technology combination leads to the least expensive product?

*W2,T2

2- A programmer is deciding between two systems S1 and S2 to run his code. For the specific program in his/her mind, system S1 uses instructions A, B, C, D, and E and system S2 uses instructions F, G, H, I, and J. S1 and S2 run at 2GHz and 1.5GHz respectively. The following table is also given:

Instruction	A	B	C	D	E	F	G	H	I	J
Count	1000	2000	2400	700	1300	800	3000	2200	300	500
Clocks consumed	2	10	7	4	8	12	1	3	8	16

2a) What is the CPI of each system?

S1	(1000*2+2000*10+2400*7+700*4+1300*8)/(1000+2000+2400+700+1300)	*7.0270
S2	(800*12+3000*1+2200*3+300*8+500*16)/(800+3000+2200+300+500)	*4.3529

2b) What is the total execution time on each system?

S1	(1000*2+2000*10+2400*7+700*4+1300*8)/(2e9)	*26 micro sec
S2	(800*12+3000*1+2200*3+300*8+500*16)/(1.5e9)	*19.73 micro sec

2c) If the CPI of instruction B and C is improved by a factor of .5, what is the new execution time on S1?

(1000*2+2000*5+2400*3.5+700*4+1300*8)/(2e9)	*16.8 micro second

2d) What is the performance improvement caused by 2c in percentage?

[(26/16.8)-1]*100	*54.76%

3- Suppose that a program consists of N instruction types $I_1,\ldots,I_N$. Instruction counts are given by $n_1, \ldots, n_N$. The clocks consumed per instruction type are $c_1,\ldots,c_N$.

3a) If each CPI is improved by a certain factor ($a_i c_i$ instead of c_i, where a_i is a number between 0 and 1), write down the equation for total performance improvement in percentage.

$$*100 * \left[\frac{c_1 n_1 + \cdots c_N n_N}{a_1 c_1 n_1 + \cdots a_N c_N n_N} - 1\right]$$

3b) If only one of the CPIs is to be improved, which instruction type should be selected to get the highest improvement?

*The one with largest c*n

3c) Assume that the index of instruction type found in 3b) is j. If this instruction can be improved without any limit (0 clock consumed as an extreme impossible case), what is the limit of overall performance improvement?

$$*100 * \left[\frac{\sum_i c_i n_i}{\sum_{i\neq j} c_i n_i} - 1\right] = 100 * \frac{c_j n_j}{\sum_{i\neq j} c_i n_i}$$

4a) Fill out the following SPEC tables. Calculate all times in micro second.

System1, 1GHz:

Program	Instruction types						Execution time	Ref1	SPEC with ref 1	Ref2	SPEC with ref 2
	A		B		C						
	Count	CPI	Count	CPI	Count	CPI					
1	547	5	648	3	945	13	*16.9640	390.1720	*23	661.5960	*39
2	426		679		210		*6.8970	158.6310	*23	124.1460	*18
3	645		636		710		*14.3630	473.9790	*33	473.9790	*33
SPEC ratio with geometric mean:								*25.9412		*28.5069	
SPEC ratio with arithmetic mean:								*26.3333		*30	

System2, 1.5GHz:

Program	Instruction types						Execution time	Ref1	SPEC with ref 1	Ref2	SPEC with ref 2
	A		B		C						
	Count	CPI	Count	CPI	Count	CPI					
1	417	7	257	3	541	19	*9.3127	390.1720	*41.8969	661.5960	*71.0426
2	842		614		870		*16.1773	158.6310	*9.8058	124.1460	*7.6741
3	833		583		266		*8.4227	473.9790	*56.2742	473.9790	*56.2742
SPEC ratio with geometric mean:								*28.4877		*31.3053	
SPEC ratio with arithmetic mean:								*35.9923		*44.9970	

4b) When comparing the performance of system 1 and 2, we can choose any combination of reference system (ref 1 or ref 2) and mean calculation method (arithmetic or geometric). This results in 4 different combinations. How much faster (in percentage) is system 2 compared to system 1 under each of the 4 possible combinations?

	Ref1		Ref2	
Arithmetic	[35.9923/26.333-1]*100	*36.68%	[44.9970/30-1]*100	*49.99%
Geometric	[28.4877/25.9412-1]*100	*9.82%	[31.3053/28.5069-1]*100	*9.82%

4c) From 4b), what do you conclude about dependence of performance comparison on reference system under arithmetic and geometric mean?

*Geometric mean removes the dependence on reference system

5- Consider the following specification of two processors:

	Clock rate	Dynamic power	Static Power	Voltage
Processor A	3.6GHz	90W	10W	1.25V
Processor B	3.4GHz	40W	30W	.9V

5a) Using the equation:

dynamic power = .5 x C x V^2 x f

where C is the effective capacitive load, calculate the effective capacitive load of each processors

Processor A	90/(.5*1.25^2*3.6e9)	*32 nF
Processor B	40/(.5*.9^2*3.4e9)	*29.0487 nF

5b) Suppose that the two processors consume the same amount of total energy per instruction for a specific program. Denote the instruction per second of processor A and B by n1 and n2. Find n1/n2.

(90+10)/n1=(40+30)/n2	*n1/n2=100/70

5c) Following 5b), calculate (CPI of A)/(CPI of B)?

n1/n2=(3.6Ghz/CPI1)/(3.4GHz/CPI2)=100/70	*CPI1/CPI2=.7412

5d) Does 5b) mean that the two processors consume the same amount of energy to run a program? Why?

*Not necessarily. We don't know the total number of instructions used by the program on each processor. Total time and thus total energy depends on instruction count as well.

5e) Suppose that the two processors use the same amount of energy to run the program. Denote total number of instructions for A and B by N1 and N2. Calculate N1/N2?

(90+10)*(N1*CPI1/F1)=(40+30)*(N2*CPI2/F2)	*N1/N2=1

6- Suppose that a specific task takes T seconds on a single core processor. When divided over n cores, the processing time on each core takes T/n seconds. However, an overhead T*(.01*sqrt(n)-.01) is added when using multiple cores.

6a) Write down the equation for total execution time as a function n.

* $\frac{T}{n} + T[.01\sqrt{n} - .01]$

6b) Use calculus to find the number of cores that lead to maximum performance. Round to the nearest integer.

Derivative: $\frac{-T}{n^2} + T[.005\frac{1}{\sqrt{n}}] = 0$	*$n = 200^{2/3} \approx 34$

6c) If the cost is given by $1000*(n^.2), what is the number of cores that leads to minimum cost per unit performance? (performance is the inverse of execution time). You can use any method to solve the equation resulting from setting the derivative to 0. www.wolframalpha.com is a very useful online engine you can use.

$\frac{d}{dn}\left[(1000n^{.2})(\frac{T}{n} + T[.01\sqrt{n} - .01])\right] = 0$	*$n = 24.502 \approx 25$