**CSC 250 – Abstraction, Technology, Performance**

**Do NOT cram your answers into a small space. Expand as needed.**

**Neatness, readability, and organization are part of the grade.**

1. **(Adapted from 1.4)** Assume a color display using 8 bits for each of the primary colors (red, green, blue) per pixel and a frame size of 1280 x 1024. Don’t discount the utility of using Excel or Google sheets for solving problems. They are incredibly useful, especially when solving the same problem for different values. For this problem, you may find it convenient to fill in the table.
2. What is the minimum size in bytes of the frame buffer to store a frame?

8\*3\*1280\*1024 / 8 = 3932160

1. How long (in hours) would it take to transmit a 2-hour movie over a 120 Mbits/second network? The movie has 30 frames for every second with bytes/frame as in part a? Think about your answer and your experience watching movies at home.

2\*60 \* 60 \* 30 \* 3932160 \* 8 / (120 \* 10^6\*3600) = 15.7286

1. What compression rate (compression ratio) would you need so that the movie could be transmitted over the 120 Mbits/second network and watched in real time? By real time, it is meant that the movie can be transmitted at 30 frames per second. Thus, the number of bytes in each frame must be reduced. The compression ratio is the number of original bytes in a frame (or in the movie) divided by the number of bytes in a frame (or in the movie). So, if hall the number of bytes is transmitted, the compression ratio is 2:1.

3932160 \* 8 \* 30 / 120M = 7.8643

1. What bandwidth in Mbits/second would you need so that the uncompressed movie could be transmitted over the network and watched in real time?

3932160 \* 8 \* 30 / 1 = 943718400

|  |  |  |
| --- | --- | --- |
| **Description** | **Value** | **Comments** |
| bits per color | 8 |  |
| colors (bytes) per pixel | 3 |  |
| bits per pixel | 24 |  |
| rows per frame | 1,024 |  |
| columns per frame | 1,280 |  |
| pixels per frame |  |  |
| Bytes per frame | 3932160 | Answer to a |
| bits per frame |  |  |
|  |  |  |
| Movie frames/second | 30 |  |
| Frames in a 2-hour movie |  |  |
| Bits in a 2-hour movie |  |  |
| Network bandwidth in Mbits/second | 120 |  |
| Seconds to transmit the movie |  |  |
| Minutes to transmit the movie |  |  |
| Hours to transmit the movie | 15.7286 | Answer to b |
| Bandwidth (uncompressed) in bits/second | 943718400 | Answer to d |
| Compression rate needed for real time transmission | 7.8643 | Answer to c |

1. **1.15.5 (These numbers refer to Chapter 1, Section 15, Exercise 5. Similarly, below.)**
   1. P1 has time 3Ghz/1.5 = 2e9, P2 has time 2.5Ghz/1=2.5e9, P3 has time 4Ghz/2.2 = 1.82e9

P2 has the highest performance.

* 1. **The number of cycle:**

for P1 is 10 \* 3 \* 10^9 = 30 \* 10^9, The number of cycle for P2 is 25 \* 10^9, and the number of cycle for P3 is 40 \* 10 ^9.

**The instruction count:**

for P1 is 20 \*10^9. The instruction count for P2 is 25\*10^9. The instruction count for P3 is 18.18 \* 10^9

* 1. Clock rate for P1 1.2 \* 3 \*10^9/0.7= 5.14Ghz. Clock rate for P2 1.2 \* 2.5Ghz/0.7=4.28Ghz. Clock rate for P3 1.2\*4Ghz/0.7= 6.85Ghz

1. **1.15.7**
   1. Global CPI for P1 is 2.6**.** Global CPI for P2 is (2\*10^5 + 2\*10^5\*2 + 2\*10^5 \* 5 + 2\*10^5\*2)/3Ghz \* 3Ghz / 1e6 = 2
   2. Clock cycles for P1 is 2.6\*1e6= 26\*1e5. Clock cycles for P2 is 2\*1e6 = 20\*1e5. CPU time for P1 is 2.6e6/2.5e09 = 1.04e-3. Time for P2 is 2.0e6/3e9 = 0.66e-3. I think P2 is faster.
2. **1.15.8 a and b**
   1. CPI for A is 1.1/1e-9/1e9 = 1.1. CPI for B is 1.5/1e-9/1.2e9 = 1.25
   2. It should be 1.2e9\*1.25/(1e9\*1.1) = 1.36 times faster.
3. **(Adapted from 1.15.9)** The Pentium 4 Prescott processor, released in 2004, had a clock rate of 3.6 GHz and voltage of 1.25 V. Assume that, on average, it consumed 10 W of static power and 90 W of dynamic power. When the processor is busy, the processor is using static and dynamic power, 100 W.

The Core i5 Ivy Bridge, released in 2012, had a clock rate of 3.4 GHz and voltage of 0.9 V. Assume that, on average, it consumed 30 W of static power and 40 W of dynamic power, thus 70 W when busy.

* 1. Assume for any time period, T, the Pentium 4 is busy 72% of this time and idle the rest. Assuming the Core i5 has the exact same instruction set and is running the same program compiled by the same compiler, what percent of the time must it be busy to accomplish the same work as the Pentium 4?

What are the assumptions here? The processors are identical except for the clock and the power. I.e., identical instruction set architectures (ISA). This implies the same CPI as well. Work is interpreted as the number instructions executed.

You want the time to be same.

(Instruction count \* CPI)/3.6Ghz \*0.72 = (Instruction count \* CPI)/3.4Ghz \* x

Solve we get x = 0.7624

Which is 76.24%

I get 76.24%

* 1. (*Optional/Advanced*) For a given time period, T, what is the ratio of the energy used of the Pentium 4 to the Core i5, assuming the work to be done as described above? I.e., the percent busy time stated and determined in part a. Assume that dynamic power does not include the static power. I.e., you must include them separately as needed.

I get 1.24 Note: energy = power x time (or, power = energy/time)

1. **1.15.13 a and b**
   1. P1 = 5e9 \* 0.9/4Ghz = 1.125. P2 = 1e9\*0.75/3Ghz = 0.25. This means despite P1 has bigger clock rate, it has worse performance.
   2. P1 CPU time = 1e9 \*0.9/4Ghz = 0.225. P2 instruction count = 0.225\*3e9 / 0.75 = 9e8. P1 does more instructions but the time are the same.
2. **1.15.14**
   1. 70\*0.2 = 14s
   2. (250-70-40-85)\*0.2 = 11s
   3. No because even if time for branch instructions is 0 the reduction rate is still 40/250 = 0.16 < 0.2
3. **1.15.15 b**

Time = (50e6 + 110e6+ 80e6 \*4 + 16e6\*2)/2Ghz = 0.256.

newTime = (50e6 + 110e6+ 80e6 \*newCPI + 16e6\*2)/2Ghz = 0.128

newCPI = 0.8. We need to improve by 500%.

1. **(Adapted from 1.15.16)** Complete the table below. Think about the answers, don't just fill in numbers.

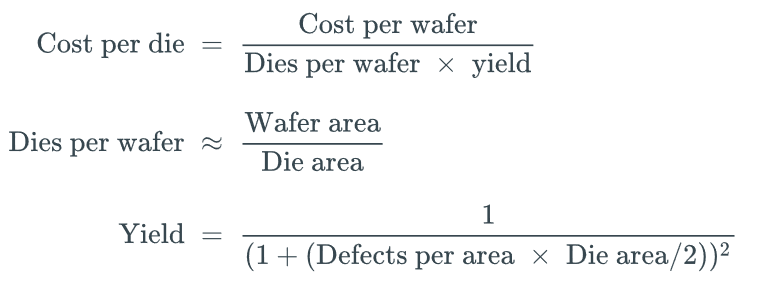
**When a program is adapted to run on multiple processors in a multiprocessor system, the execution** time on each processor is comprised of computing time and the overhead time (for example, required for locked critical sections and/or to send data from one processor to another).

Assume a program requires t = 201 s of execution time on one processor. When run on p processors, each processor requires t/p s, plus an additional 5.1 s regardless of the number of processors. Compute the per-processor execution time for 2, 4, 8, 16, 32, 64, and 128 processors. For each case, list the corresponding speedup relative to a single processor and the ratio between actual speedup versus ideal speedup (speedup if there was no overhead). Note: ideal speedup would be p. Efficiency is actual speedup divided by ideal speedup.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| 1 | 201.00 |  |  |  |
| 2 | 100.50 | 105.60 | 1.90 | 0.95 |
| 4 | 50.25 | 55.35 | 3.63 | 0.91 |
| 8 | 25.13 | 30.23 | 6.65 | 0.83 |
| 16 | 12.56 | 17.66 | 11.38 | 0.71 |
| 32 | 6.28 | 11.38 | 17.66 | 0.55 |
| 64 | 3.14 | 8.24 | 24.4 | 0.38 |
| 128 | 1.57 | 6.67 | 30.13 | 0.23 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Processors** | **exec time/proc w/o overhead** | **exec time w/overhead** | **speedup** | **actual/ideal (efficiency)** |
| 1 | 201.00 |  |  |  |
| 2 | 100.50 | 105.60 | 1.90 | 0.95 |
| 4 | 50.25 | 55.35 | 3.63 | 0.91 |
| 8 | 25.13 | 30.23 | 6.65 | 0.83 |
| 16 | 12.56 | 17.66 | 11.38 | 0.71 |
| 32 | 6.28 | 11.38 | 17.66 | 0.55 |
| 64 | 3.14 | 8.24 | 24.39 | 0.38 |
| 128 | 1.57 | 6.67 | 30.13 | 0.24 |

1. **1.15.11 a and b** Note: In the formula for *Yield*, "Defects per area" in the denominator means defects per cm2. Here are formulas from the text. Doing this in a spreadsheet might be a good approach.



* 1. 1st wafer Yield = 1/(1+(0.02 \* 7.5^2\*pi / 84/2 ))^2 = 0.9592. 2nd wafer Yield =1/(1+(0.031 \* 10^2\*pi / 100/2 ))^2 = 0.9093
  2. 1st wafer cost per die = 12/(84 \* 0.9592) = 0.1489. 2nd wafer cost per die = 15/(100\*0.9093)= 0.1649

1. Using your own PC, Laptop, whatever (or that of your friend/roommate/alien/…), find out what you can about: processor type, number of cores, clock rate, amount of main memory (RAM), cache (we will cover this later in the semester), cycles per instruction (or instructions per cycle), power consumption, anything else of interest. Work and discuss this with a classmate or another CS major or minor.

I worked and/or discussed this question with: ­­\_\_a classmate\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

I learned something I did not know or understand (circle one of the below that applies the most).

* For sure
* Kind of
* Not really but maybe
* No

**Briefly comment here what you learned or why you didn’t learn anything.**

Processor: AMD Ryzen 5 5600u

Number of cores:6

Clock rate: 2.3Ghz

RAM: 16GB

Cache: L1:384kb, L2:3mb, L3:16mb

I learned about what is cache and its difference between register.