Homework 1

Name: Aina Azman
Net_ID: qistina

1.1 Aside from the smart cell phones used by a billion people, list and describe four other types of computers.

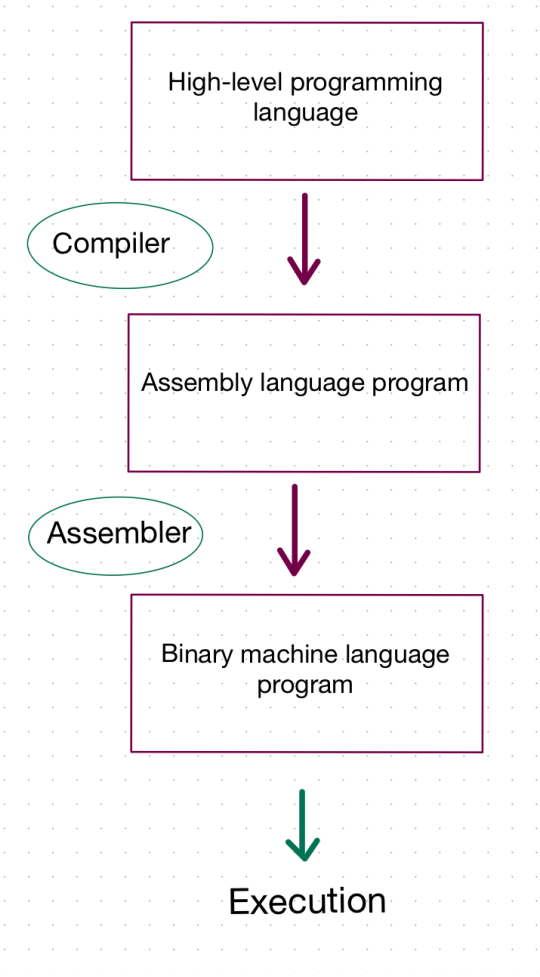
1. Personal Computer (PC): A computer designed for use by individuals, usually incorporating a graphics display, a keyboard, and a mouse.
2. Server: A computer used for running larger programs for multiple users, often simultaneously, and typically accessed only via a network.
3. Embedded Computers: A computer inside another device used for running one predetermined application or collection of software.
4. Supercomputer: A class of computers with the highest performance and cost; they are configured as servers and typically cost tens to hundreds of millions of dollars.

1.3 Describe the steps that transform a program written in a high-level language such as C into representation that is directly executed by a computer processor.

1. Firstly, programmers express their instructions by writing their program using high-level programming languages, benefiting from the human-readable syntax and abstractions.
2. The high-level source code is processed by a compiler.The compiler is a specialized program that translates the entire program written in the high-level language into an intermediate code or better known as assembly language.
3. Assembly language is a low-level symbolic representation that is still human readable and closely corresponds to the architecture of the computer. Each line in the assembly code typically corresponds to one machine-level language.
4. The assembly language program is then processed by an assembler. The assembler converts each assembly language instruction into its equivalent binary machine code. It replaces symbolic instructions with the actual binary patterns that the computer's central processing unit (CPU) can directly execute.
5. The output of the assembler is a binary machine language program. This program consists of a sequence of binary instructions that the computer processor can directly execute. Each instruction corresponds to specific operations that the CPU can perform.

6. The binary machine language program is later loaded into the computer's memory, and the CPU executes the instructions sequentially.

Conclusion:



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.

a. What is the minimum size in byte of the frame buffer to store a frame?

- Total number of bits per pixel: $3 \times 8 = 24\ bits$
- Total number of pixels in the frame: 1280×1024
- Total number of bits required to store a frame: $1280 \times 1024 \times 24$
- Total bytes needed: $\frac{1280 \times 1024 \times 24}{8} = 3,\ 932,\ 160\ bytes$
- Hence, the minimum size in byte of the frame buffer needed to store a frame is $3,\ 932,\ 160\ bytes$.

b. How long would it take, at a minimum, for the frame to be sent over 100 Mbit/s network?

- Time taken: $\frac{3,932,160 \times 8}{100 \times 10^6} = 0.3146\ seconds$
- Hence the minimum time taken for the frame to be sent over 100 Mbit/s network is approximately $0.3146\ seconds$.

1.5 Consider three different processors P1, P2, and P3 executing the same instruction set:

- P1 has 3GHz clock rate and CPI of 1.5.
- P2 has a 2.5GHz clock rate and CPI of 1.0.
- P3 has a 4.0GHz clock rate and has a CPI of 2.2.

a. Which processor has the highest performance expressed in instructions per second?

$$IPS = \frac{Instruction}{Second} = \frac{Instruction}{Second} \times 1 = \frac{Instruction}{Second} \times \frac{Clock\ Cycles}{Clock\ Cycles} = \frac{Instruction}{Clock\ Cycles} \times \frac{Clock\ Cycles}{Second}$$

$$CPI = \frac{CPU\ clock\ cycles}{Instruction} \quad Clock\ Rate = \frac{Clock\ Cycles}{Seconds}$$

$$IPS = \frac{1}{CPI} \times Clock\ Rate, \quad Hence\ IPS = \frac{Clock\ Rate}{CPI}$$

- $IPS_{P1} = \frac{3 \times 10^9}{1.5} = 2GHz$
- $IPS_{P2} = \frac{2.5 \times 10^9}{1.0} = 2.5\ GHz$
- $IPS_{P3} = \frac{4.0 \times 10^9}{2.2} = 1.818\ GHz$

Hence, P2 has the highest performance expressed in instructions per second.

b. If the processor each executes a program in 10 seconds, find the number of cycles and the number of instructions.

$$CPU\ time = \frac{CPU\ clock\ cycles}{clock\ rate}, \quad CPU\ clock\ cycles = Instructions\ for\ a\ program \times CPI$$

$$Number\ of\ cycles = CPU\ time \times Clock\ rate, \quad Instructions\ for\ a\ program = \frac{clock\ cycles}{CPI}$$

- $Number\ of\ cycles_{P1} = 10 \times 3.0GHz = 30\ billion\ cycles$

  $Number\ of\ instructions_{P1} = \frac{30\ billion}{1.5} = 20\ billion\ instructions$

- $Number\ of\ cycles_{P2} = 10 \times 2.5GHz = 25\ billion\ cycles$

  $Number\ of\ instructions_{P2} = \frac{25\ billion}{1.0} = 25\ billion\ instructions$

- $Number\ of\ cycles_{P3} = 10 \times 4.0GHz = 40\ billion\ cycles$

  $Number\ of\ instructions_{P3} = \frac{40\ billion}{2.2} = 18.182\ billion\ instructions$

c. We are trying to reduce the execution time by 30%, but this leads to an increase of 20% in the CPI.What clock rate should we have to get this time reduction?

$$CPU\ time = \frac{Instruction\ \ Count \times CPI}{Clock\ Rate}$$

$$Clock\ Rate = \frac{Instruction\ \ Count \times CPI}{CPU\ time}$$

- $Clock\ Rate_{P1} = \frac{(20\times10^{9}) \times (1.5 \times 1.2)}{10 \times 0.7} = 5.14\ GHz$
- $Clock\ Rate_{P2} = \frac{(25\times10^{9}) \times (1.0 \times 1.2)}{10 \times 0.7} = 4.29\ GHz$
- $Clock\ Rate_{P1} = \frac{(18.182\times10^{9}) \times (2.2 \times 1.2)}{10 \times 0.7} = 6.86\ GHz$

1.7 Compilers can have a profound impact on the performance of an application. Assume that for a program

- Compiler A results in a dynamic instruction count of 1.0E9 and has an execution time of 1.1s
- While compiler B results in a dynamic instruction count of 1.2E9 and an execution time of 1.5s

a. Find the average CPI for each program given that the processor has a clock cycle time of 1ns.

$$Clock\ Rate = \frac{1}{clock\ cycle\ time}, \qquad Clock\ Rate = \frac{1}{1\times10^{-9}} = 1GHz$$

$$CPU\ time = \frac{Instruction\ \ Count \times CPI}{Clock\ Rate}$$

- $CPI_{A} = \frac{1.1\times 1\ GHz}{1\times10^{9}} = 1.1$

- $CPI_B = \frac{1.5 \times 1\ GHz}{1.2 \times 10^9} = 1.25$

b. Assume that the compiled run on two different processors. If the execution time on the two processors are the same, how much faster is the clock of the processor running compiler A's code versus the clock of the processor running compiler B's code?

$$CPU\ time = \frac{Instruction\ \ Count \times CPI}{Clock\ Rate}$$

$$CPU\ time_A = CPU\ time_B$$

$$\frac{Clock\ Rate_A}{Clock\ \ Rate_B} = \frac{1 \times 10^9 \times 1.1}{1.2 \times 10^9 \times 1.25}$$

$$= \frac{11}{15} = 0.73s$$

Hence, the clock of the processor running compiler A is 0.73 times faster than compiler B.

c. A new compiler is developed that uses only 6.0E8 instructions and has an average CPI of 1.1. What is the speedup of using this new compiler versus using a compiler A or B on the original processor?

$CPU\ time = Instruction\ count \times CPI \times Clock\ Cycle\ Time$

$CPU\ \ time_c = 6.0 \times 10^8 \times 1.1 \times 10^{-9} = 0.66\ s$

- $Speedup_{A-C} = \frac{1.1}{0.66} = 1.67,\ Hence\ the\ new\ compiler\ is\ 1.67\ faster\ than\ compiler\ A$
- $Speedup_{B-C} = \frac{1.5}{0.66} = 2.27,\ Hence\ the\ new\ compiler\ is\ 2.27\ faster\ than\ compiler\ B$

1.10

Wafer 1: Assume a 15 cm diameter wafer has a cost of 12, contains 84 dies, and has 0.020 defects/cm^2.

Wafer 2: Assume a 20 cm diameter wafer has a cost of 15, contains 100 dies, and has 0.031 defects/cm^2.

1.10.1 [10] <§1.5> Find the yield for both wafers.

$$Yield = \frac{1}{(1 + (Defects\ per\ area \times Die\ area/2))^2}$$

- $Area_1 = \pi\left(\frac{15}{2}\right)^2 = \frac{225}{4}\pi$

  $Die\ area_1 = \frac{225\pi}{4} \div 84 = \frac{75\pi}{112}$

  $Yield_1 = \frac{1}{(1 + (0.02 \times \frac{75\pi}{112}/2))^2} = 0.979$

- $Area_2 = \pi\,(10)^2 = 100\pi$

  $Die\ area = 100\pi \div 100 = \pi$

  $Yield_2 = \frac{1}{(1 + (0.031 \times \pi/2))^2} = 0.909$

1.10.2 [5] <§1.5> Find the cost per die for both wafers.

$Cost\ per\ die = \frac{Cost\ per\ wafer}{Dies\ per\ wafer \times yield}$

- $Cost\ per\ die_1 = \frac{12}{84 \times 0.979} = 0.146$
- $Cost\ per\ die_2 = \frac{15}{100 \times 0.909} = 0.165$

1.10.3 [5] <§1.5> If the number of dies per wafer is increased by 10% and the defects per area unit increases by 15%, find the die area and yield.

- $Number\ of\ dies_1 = 84 \times 1.1 = 92$

  $Defects\ per\ area_1 = 0.02 \times 1.15 = 0.023$

  $Die\ area_1 = \frac{225\pi}{4} \div 92 = 1.921$

  $Yield_1 = \frac{1}{(1 + (0.023 \times 1.921/2))^2} = 0.957$

- $Number\ of\ dies_2 = 100 \times 1.1 = 110$

  $Defects\ per\ area_2 = 0.031 \times 1.15 = 0.036$

  $Die\ area_2 = 100\pi \div 110 = 2.856$

$$Yield_2 = \frac{1}{(1 + (0.036 \times 2.856/2))^2} = 0.905$$

1.10.4 [5] <§1.5> Assume a fabrication process improves the yield from 0.92 to 0.95. Find the defects per area unit for each version of the technology given a die area of 200 mm2.

$$Yield = \frac{1}{(1 + (Defects\ per\ area \times Die\ area/2))^2}$$

$$Defects\ per\ area = (\sqrt{\frac{1}{Yield}} - 1) \div \frac{Die\ Area}{2}$$

- $Defects\ per\ area_{0.92} = (\sqrt{\frac{1}{0.92}} - 1) \div \frac{200}{2} = 4.257 \times 10^{-4}$
- $Defects\ per\ area_{0.95} = (\sqrt{\frac{1}{0.95}} - 1) \div \frac{200}{2} = 2.598 \times 10^{-4}$

1.12 Section 1.10 cites as a pitfall the utilization of a subset of the performance equation as a performance metric. To illustrate this, consider the following two processors:

- P1 has a clock rate of 4GHz, average CPI of 0.9, and requires the execution of 5.0E9 instructions.
- P2 has a clock rate of 3GHz, an average CPI of 0.75, and requires the execution of 1.0E9 instructions.

1.12.1 [5] <§§1.6, 1.10> One usual fallacy is to consider the computer with the largest clock rate as having the highest performance. Check if this is true for P1 and P2.

$$CPU\ time = Instruction\ count \times CPI \times Clock\ Cycle\ Time$$

- $CPU\ time_{P1} = 5.0 \times 10^9 \times 0.9 \times \frac{1}{4 \times 10^9} = 1.125\ s$
- $CPU\ time_{P2} = 1.0 \times 10^9 \times 0.75 \times \frac{1}{3 \times 10^9} = 0.25\ s$
- Hence, the conclusion of the computer with the largest clock rate having the highest performance is false. The CPU time for P2 is shorter than the P1. This means that the performance of P2 is greater than P1.

1.12.2 [10]<§§1.6,1.10>Another fallacy is to consider that the processor executing the largest number of instructions will need a larger CPU time. Considering that:

Processor P1 is executing a sequence of 1.0E9 instructions and that the CPI of processors P1 and P2 do not change, determine the number of instructions that P2 can execute in the same time that P1 needs to execute 1.0E9 instructions.

$$CPU\ time = Instruction\ count \times CPI \times Clock\ Cycle\ Time$$

- $CPU\ time_{P1} = 1.0 \times 10^9 \times 0.9 \times \frac{1}{4 \times 10^9} = 0.9\ seconds$
- $CPU\ time_{P2} = 0.9 = Instruction\ Count \times 0.75 \times \frac{1}{3 \times 10^9}$

  $Instruction\ count_{P2} = \frac{0.9}{2.5 \times 10^{-10}} = 3.6E9$

- Hence, the conclusion that the processor executing the largest number of instructions will need a larger CPU time is false. The CPU time for P1 and P2 is the same yet P2 executes more instructions than P1.

1.12.3 [10] <§§1.6, 1.10> A common fallacy is to use MIPS (millions of instructions per second) to compare the performance of two different processors, and consider that the processor with the largest MIPS has the largest performance. Check if this is true for P1 and P2.

$$MIPS = \frac{Instruction\ Count}{Execution\ time \times 10^6}$$

- $MIPS_{P1} = \frac{5 \times 10^9}{1.125 \times 10^6} = 4444.4444$
- $MIPS_{P2} = \frac{1 \times 10^9}{0.25 \times 10^6} = 4000$
- The MIPS of P1 is greater than P2. But, referring again to the CPU time, P2's CPU time is shorter than P1. Performance of the processor is evaluated through the CPU time. Hence, to consider the processor with the largest MIPS has the largest performance is wrong.

1.12.4 [10] <§1.10> Another common performance figure is MFLOPS (millions of floating-point operations per second), defined as

MFLOPS = No. FP operations /(execution time X 1E6)

but this figure has the same problems as MIPS. Assume that 40% of the instructions executed on both P1 and P2 are floating-point instructions. Find the MFLOPS figures for the processors.

- $FP_1 = 0.4 \times 5 \times 10^9 = 2E9$

  $MFLOPS_1 = \frac{2 \times 10^9}{1.125 \times 10^6} = 1777.7778$

- $FP_2 = 0.4 \times 1 \times 10^9 = 0.4E9$

  $MFLOPS_2 = \frac{0.4 \times 10^9}{0.25 \times 10^6} = 1600$

1.14 Assume a program requires the execution of 50 × 10^6 FP instructions, 110 × 106 INT instructions, 80 × 10^6 L/S instructions, and 16 × 10^6 branch instructions. The CPI for each type of instruction is 1, 1, 4, and 2, respectively. Assume that the processor has a 2 GHz clock rate.

1.14.1 [10] <§1.10> By how much must we improve the CPI of FP instructions if we want the program to run two times faster?

$$CPU\ time\ =\ \frac{Instruction\ count \times CPI}{Clock\ Rate}$$

$$CPU\ time\ =\ \frac{50 \times 10^6 \times 1}{2 \times 10^9}\ =\ 0.025s$$

$$CPU\ time_{\times 2} = \frac{50 \times 10^6 \times CPI}{2 \times 10^9}\ =\ 0.05s \qquad CPI\ =\ \frac{0.05 \times 2 \times 10^9}{50 \times 10^6}\ =\ 2$$

Hence, the CPI of FP must be 2 if we want the program to run two times faster.

1.14.2 [10] <§1.10> By how much must we improve the CPI of L/S instructions if we want the program to run two times faster?

$$CPU\ time\ =\ \frac{80 \times 10^6 \times 4}{2 \times 10^9}\ =\ 0.16s$$

$$CPU\ time_{\times 2} = \frac{80 \times 10^6 \times CPI}{2 \times 10^9}\ =\ 0.32s \qquad CPI\ =\ \frac{0.32 \times 2 \times 10^9}{80 \times 10^6}\ =\ 8$$

Hence, the CPI of L/S must be 8 if we want the program to run two times faster.

1.14.3 [5] <§1.10> By how much is the execution time of the program improved if the CPI of INT and FP instructions is reduced by 40% and the CPI of L/S and Branch is reduced by 30%

$$CPU\ time_{original}\ =\ \frac{((50\times 10^6) + (110 \times\ 10^6)\ +(80 \times 10^6 \times 4) + (16 \times 10^6 \times 2))}{2 \times 10^9}\ =\ 0.256\ s$$

$$CPU\ time_{improved}\ =\ \frac{((50\times 10^6 \times 0.6) + (110 \times\ 10^6 \times 0.6)\ +(80 \times 10^6 \times 4 \times 0.7) + (16 \times 10^6 \times 2 \times 0.7))}{2 \times 10^9}\ =\ 0.171\ s$$

$$\frac{Execution\ Time_{original}}{Execution\ Time_{improved}}\ =\ \frac{0.256\ s}{0.171s}\ =\ 1.497$$

Hence, the CPU time of the improved program is 1.497 faster than the CPU time of the original.