

**Computer Basics**

**FIT  
100**

Electronic computers have changed dramatically over their 50 history, but a few basic principles characterize all computers

© Copyright 1999-2000 University of Washington

**FIT  
100**

**Abstractly, A Computer Is ...**

- ❖ Computers process information by deterministically following (executing) instructions
- ❖ Unlike humans, computers follow instructions *exactly*
  - ❑ Computers have no imagination or creativity
  - ❑ Computers have no intuition
  - ❑ Computers are literal, with no sense of irony, subtlety, proportion, ...
  - ❑ Computers don't joke, they're not vindictive or cruel
  - ❑ Computers are not purposeful
- ❖ ... computers only execute instructions

If a computer has any useful characteristics, it's because someone has programmed it – given it the instructions – to behave usefully

© Copyright 1999-2000 University of Washington

**FIT  
100**

**Interpreting Instructions**

- ❖ To perform instructions, computers use a process known as the *fetch/execute cycle* implemented in their hardware

**Fetch/Execute Cycle**

- Instruction Fetch (IF)
- Instruction Decode (DE)
- Data Fetch (DF)
- Instruction Execution (EX)
- Return Result (RR)

- ❖ The F/E Cycle is an unending process, hence the red arrow

© Copyright 1999-2000 University of Washington

**FIT  
100**

**An Analogy ...**

- ❖ At the Nenana Ice Classic, where people pay \$2 to guess when the ice will break up on the Tananah River, someone processes the guesses

...  
 4 May 12:05 PM  
 4 May 12:04 PM  
 4 May 12:03 PM  
 ...

Get next entry  
 Find date/time  
 Get card for d/t  
 Enter Name  
 Return Card

Hello, Nenana!  
 I'm sure breakup will be on  
 \*\*\* May 4 at 12:04PM \*\*\*  
 Find \$2 enclosed.  
 Yours, Frost T. Snowman  
 61 River St.  
 Circle AK

© Copyright 1999-2000 University of Washington

**FIT  
100**

**Anatomy Of A Computer**

- ❖ A computer is composed five components ...
  - ❑ Arithmetic/Logic Unit (ALU) -- the part that "computes", e.g. +
  - ❑ Control -- the part that follows the Fetch/Execute Cycle of the program and tells the ALU what to compute
  - ❑ Memory -- where data, programs are kept while computing
  - ❑ Input -- ports to peripheral devices from which data comes
  - ❑ Output -- ports to peripheral devices to which data goes

© Copyright 1999-2000 University of Washington

**FIT  
100**

**Memory**

- ❖ The memory is passive, storing programs and data

0	1	2	3	4	5	6	7
R	2	D	2	100	*	R	W

- ❖ Memory is called RAM for "random access memory" because the control can access any random location in the memory
- ❖ RAM is volatile, meaning it disappears when the power is turned off ... how does the computer remember the date?
- ❖ For the control to execute (run) a program, it must be stored in the RAM. So, one operating system duty is to move programs & data from the disk to the RAM

© Copyright 1999-2000 University of Washington

## FIT 100 Control Rules!

- ❖ The control follows instructions, telling the other parts what to do
- ❖ The instructions come from the program stored in the memory

Programmers write the instructions (programs) using languages (C, C++, Java, etc.) that are way too complicated for the control to follow ... so the programs are translated into a simpler form called machine language that the control can understand. A typical machine instruction is

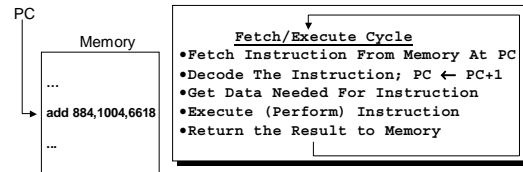
add 884, 1004, 6618

which means "add the number in memory location 884 to the number in memory location 1004, and put the result in memory location 6618"

© Copyright 1999-2000 University of Washington

## FIT 100 Following Instructions

- ❖ The control keeps track of where it is in the program using a program counter or PC ... a better name would be "instruction pointer"
- ❖ The control also fetches data and returns results to the memory



© Copyright 1999-2000 University of Washington

## FIT 100 The PC's PC

- ❖ After the instruction has been fetched from memory, the PC is incremented to refer to the next instruction in sequence
- ❖ This scheme should cause the computer to "run" through memory executing all instructions once and "fall off the end of memory"
- ❖ Computers have machine instructions to branch and jump, i.e. go to some instruction other than the next
- ❖ Jump and Branch change the PC after increment
- ❖ Programs generally repeat many instructions

3 PCs: personal computer, program counter and printed circuit board

© Copyright 1999-2000 University of Washington

## FIT 100 Emphasis ...

- ❖ The instruction add 884, 1004, 6618
- ❖ Does not add 884 and 1004 together -- we can figure that out with a calculator ... it adds whatever numbers are stored in those memory locations
- ❖ Different numbers in those locations produce different results

add 884, 1004, 6618

884	+	1004	→	6618
40		2		42
884	+	1004	→	6618
9		-2		7
884	+	1004	→	6618
699		0		699

© Copyright 1999-2000 University of Washington

## FIT 100 The Numbers, Please

- ❖ A computer memory location can store a byte of information (8 bits), enough for a keyboard character
- ❖ A "normal" whole number (integer) uses 4 bytes
- ❖ A machine instruction uses 4 bytes
- ❖ Units of memory size are ...
  - ❑ KB, kilobyte, 1,024 bytes ... just over a thousand bytes, a "K"
  - ❑ MB, megabyte, 1,048,576 bytes ... just over a million bytes, a "meg"
  - ❑ GB, gigabyte, 1,073,741,824 bytes ... just over a billion bytes, a "gig"
  - ❑ TB, terabyte, 1,099,511,627,776 bytes ... just over a trillion bytes

© Copyright 1999-2000 University of Washington

## FIT 100 Strange Numbers

- ❖ Why do computers use such strange numbers???
  - ❑ These numbers are powers of 2
    - +  $2^{10} = 1,024$  call it a thousand
    - +  $2^{20} = 1,048,576$  call it a million
    - +  $2^{30} = 1,073,741,824$  call it a billion
    - +  $2^{40} = 1,099,511,627,776$  call it a trillion
- ❖ When you buy a megabyte of memory you get 48,576 bytes for free!

© Copyright 1999-2000 University of Washington

**FIT  
100****The Pace Of Computing**

- ❖ Think of the clock rate of a computer as the rate it executes instructions, that is, how many Fetch/Execute cycles it can complete in a second (modern computers are *very* complex and can complete more than one instruction per cycle)
- ❖ hertz measures "cycles per second"
- ❖ 100MHz, specifies "100 million cycles per second"

A higher clock rate may not result in a faster running program, because the speed may be limited by other parts of the computer besides instruction execution rate; the speed of getting an instruction's data is often a limitation that worsens with a faster clock

© Copyright 1999-2000 University of Washington

**FIT  
100****Summary**

- ❖ Computers deterministically execute instructions to process information
- ❖ Computers have five parts: ALU, Control, Memory, Input and Output
- ❖ The control implements a process called the Fetch/Execute Cycle
- ❖ The fetch/execute cycle is a fundamental method of deterministically performing operations, and the idea is used many places in computation ...
  - + The computer is an electronic fetch/execute cycle, ie, hardware
  - + All other F/E cycles are implemented as programs, ie, software

© Copyright 1999-2000 University of Washington