A Brief History of Computer Science

4700 HUNDRED YEARS AGO…………

Sumerians invented the abacus

Sand, lines, pebbles

Sexagesimal

Base 60 still used today

Time, distance

How do you count like that?

SIDE TRIP…………

Factors of 60 are ….

1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30, 60

60 is a *Superior Highly Composite Number*

(number theory stuff, more factors than other numbers scaled relative to itself)

So What???

With that many factors, fractions are simplified

Example: Divide an hour into many even parts

SIDE TRIP…………

It is also a *Colossally Abundant Number*

(instead of factors, lots of divisors)

SIDE TRIP…………

1	2	
2	6	3
3	12	2
4	60	5
5	120	2
6	360	3
7	2520	7
8	5040	2
9	55440	11
10	720720	13
11	1441440	2
12	4324320	3
13	21621600	5
14	367567200	17
15	6983776800	19

Back on track…………
2100 years ago

Antikythera mechanism invented

Analog computer - what is that?

17TH CENTURY

John Napier

Logarithms - used for what

Blaise Pascal

Pascalines –

for his father

19TH CENTURY

- Charles Babbage
  - Analytic engine - for what
    - Expandable memory
    - Arithmetic unit
    - Logic unit, loops, conditional branching

19TH CENTURY

Augusta Ada King, Countess of Lovelace

Father was ?

Known as Ada Lovelace

Translated an Italian Engineer's paper

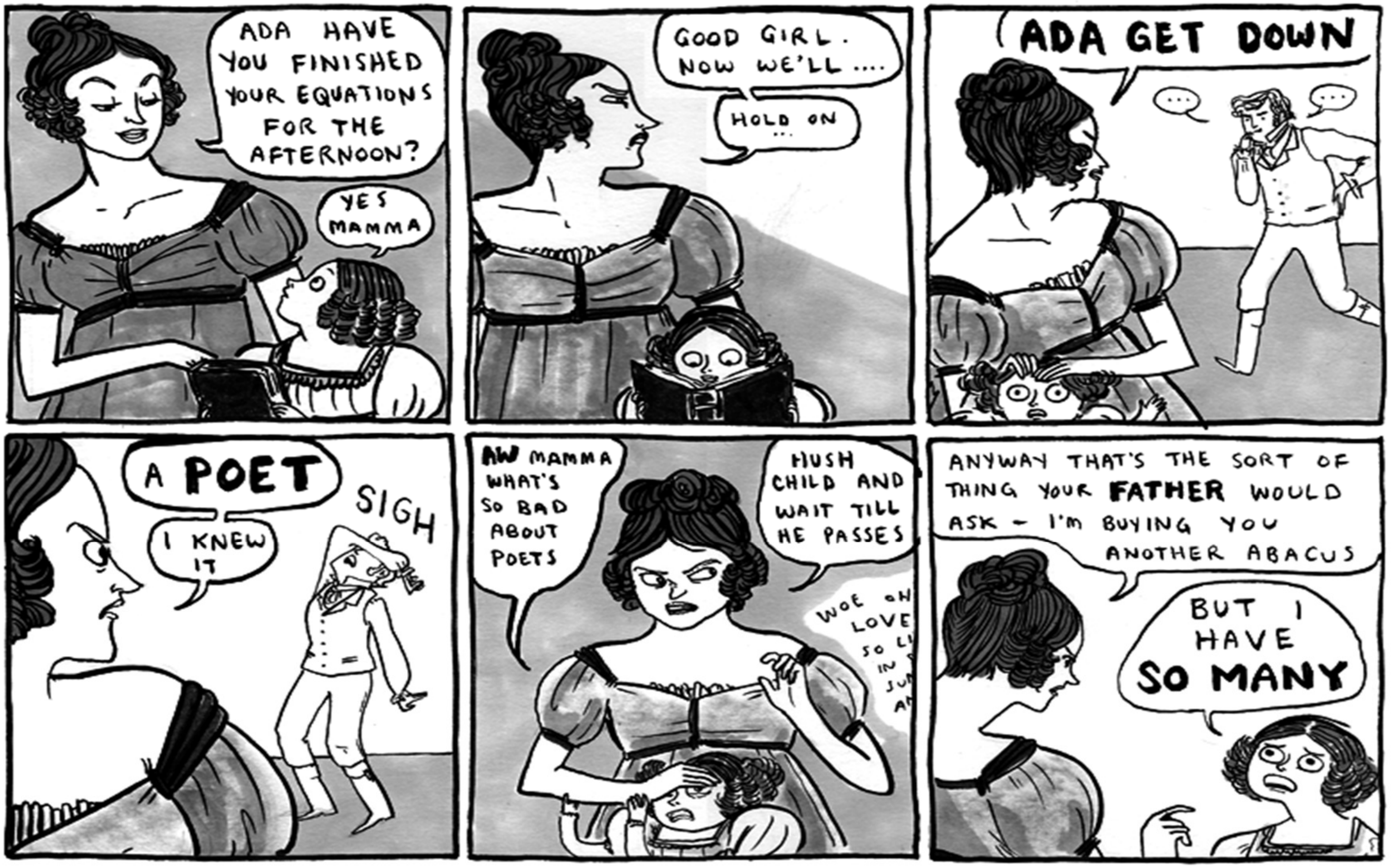
(on the analytic engine)

Wrote notes longer than the paper

Note G recognized as the world's first

algorithm (for computing Bernoulli numbers)

YOUNG ADA LOVELACE


ADA HAVE YOU FINISHED YOUR EQUATIONS FOR THE AFTERNOON?
YES MAMMA
GOOD GIRL. NOW WE'LL ....
HOLD ON ...
ADA GET DOWN
...
...
A POET
I KNEW IT
SIGH
AW MAMMA WHAT'S SO BAD ABOUT POETS
HUSH CHILD AND WAIT TILL HE PASSES
WOE OH LOVE SO LI IN JU AN
ANYWAY THAT'S THE SORT OF THING YOUR FATHER WOULD ASK - I'M BUYING YOU ANOTHER ABACUS
BUT I HAVE SO MANY

19TH CENTURY

- George Boole –
  - Boolean Algebra –
  - aided in **modeling** computations
- Joseph Jacquard
  - Loom
  - Punched card
- Herman Hollerith (Computing-Tabulating-Recording)

20TH CENTURY

Before 1920 **computers** were human beings doing complex calculations for Commerce, Government and Research

After 1920 **computing machine** began to refer to hardware that replaced human computers

Mathematicians began analyzing these kinds of machines

20TH CENTURY

Church-Turing hypothesis

"Mathematical method was 'effective' if it could be set out as a set of instructions that a human could follow with pencil and paper”

By the late 1940s machines did that

20TH CENTURY

Alan Mathison Turing

Invented a hypothetical device (model) used to study computing and computers

- The Turing machine

This is really to model computation, not to model computers

Computer science is really mostly about computation and understanding it

20th Century

The Universal Turing Machine

Can model all Turing machines

This is considered by most to be the fundamental theoretical breakthrough in computer science

The stored program computer evolved from this notion of a Universal Turing Machine

20TH CENTURY

Jon Von Neumann

Father of the computer

Renaissance guy, polyglot

Data and programs can be the same thing and all stored in memory

Computers are made up of parts

- Main memory, accumulator, ALU

End of Part 1

And you still haven’t written
a single line of code

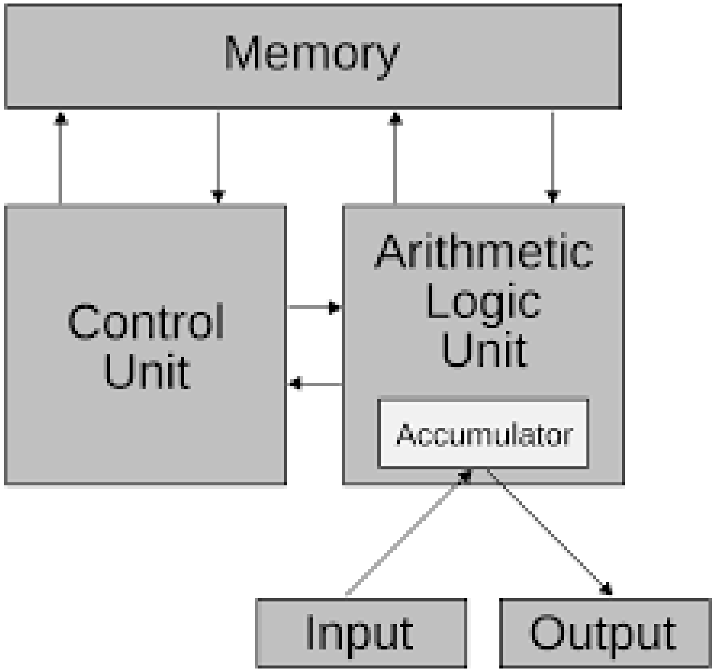
HARDWARE & SOFTWARE

HW – physical
SW – not physical

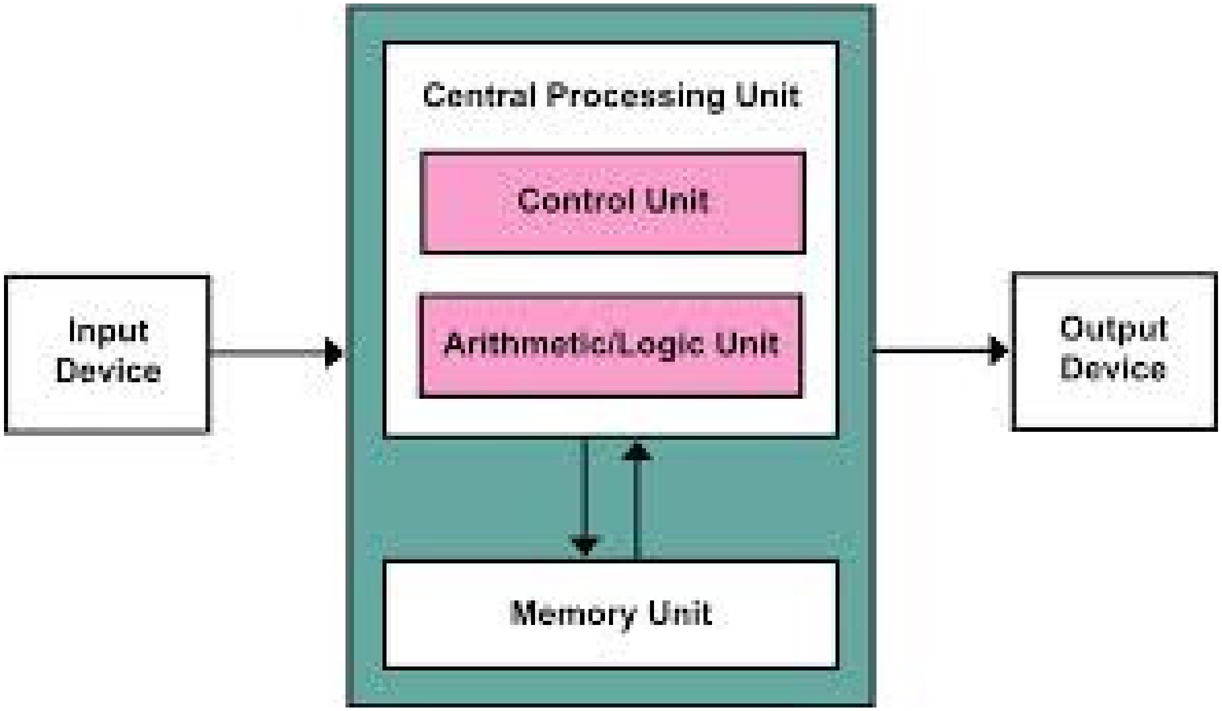
Memory –

- binary digits – bits – a bit can be 0 or 1
- byte – 8 bits

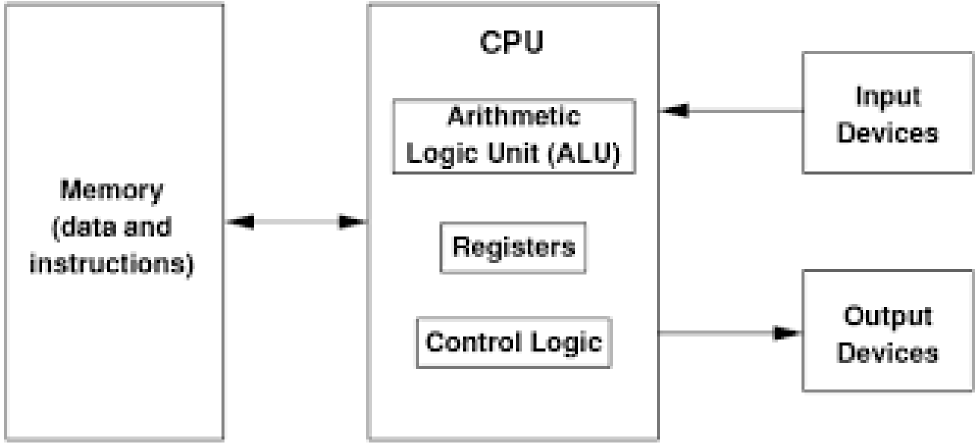
HARDWARE


Memory
Control Unit
Arithmetic Logic Unit
Accumulator
Input
Output

HARDWARE


Central Processing Unit
Control Unit
Arithmetic/Logic Unit
Input Device
Output Device
Memory Unit

HARDWARE



Memory
(data and
instructions)
CPU
Arithmetic
Logic Unit (ALU)
Registers
Control Logic
Input
Devices
Output
Devices

HARDWARE

1. User Interface
2. Auxilary Input/Output (I/O) Devices
3. Auxiliary Storage Devices
4. Network Connection
5. Internal Memory
6. Central Processing Unit

SOFTWARE

1. System Software
    1. Operating System
    2. Network Operating System
    3. Compilers/Interpreters
    4. User Interface
2. Applications Software
    1. ‘Office’ (examples)
    2. Multimedia
    3. Database
    4. Everything we write

COUNTING

1. First, let’s count
2. Base 10

   0,1,2,3,4,5,6,7,8,9
3. Base 2

   0,1
4. Base 5

   0,1,2,3,4
5. Base 8

   0,1,2,3,4,5,6,7
6. Base 16

   0,1,2,3,4,5,6,7,8,9,……….

COUNTING

1. Since a byte is 8 bits…….
2. Possible numbers?
3. Negative numbers?

COUNTING

Base 10
decimal
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34

COUNTING

Base 10	**Base 2**
decimal	binary
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111
16	10000
17	10001
18	10010
19	10011
20	10100
21	10101
22	10110
23	10111
24	11000
25	11001
26	11010
27	11011
28	11100
29	11101
30	11110
31	11111
32	100000
33	100001
34	100010

COUNTING

Base 10	**Base 2**	**Base 8**
decimal	binary	octal
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	10
9	1001	11
10	1010	12
11	1011	13
12	1100	14
13	1101	15
14	1110	16
15	1111	17
16	10000	20
17	10001	21
18	10010	22
19	10011	23
20	10100	24
21	10101	25
22	10110	26
23	10111	27
24	11000	30
25	11001	31
26	11010	32
27	11011	33
28	11100	34
29	11101	35
30	11110	36
31	11111	37
32	100000	40
33	100001	41
34	100010	42

COUNTING

Base 10	**Base 2**	**Base 8**	**Base 16**
decimal	binary	octal	hexadecimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13
20	10100	24	14
21	10101	25	15
22	10110	26	16
23	10111	27	17
24	11000	30	18
25	11001	31	19
26	11010	32	1A
27	11011	33	1B
28	11100	34	1C
29	11101	35	1D
30	11110	36	1E
31	11111	37	1F
32	100000	40	20
33	100001	41	21
34	100010	42	22

CHARACTERS

1. Need to manipulate more than numbers
2. How do you represent characters?
3. ASCII – 8 bits

   256 combinations
4. Unicode – 16 bits

   65,536 combinations

‘Other’ Stuff

1. sound
2. images
3. Video
    1. Compression
    2. Standards?
4. What else needs to be modeled?
    1. Program instructions
5. Memory

PROGRAMMING LANGUAGES

1. 1st Generation (1940s-1950s)
    1. Machine Language
2. 2nd Generation (1950s-present)
    1. Assembly Languages
3. 3rd Generation (1950s-present)
    1. High Level Languages
        1. FORTRAN, COBOL, BASIC, C, PL/1
        2. Pascal, HAL/S
        3. C++, Smalltalk, Python, Java
    2. Compiler/Interpreter

SW DEVELOPMENT LIFE CYCLE

1. Waterfall model
    1. Analysis
    2. Design
    3. Implementation
    4. Integration
    5. Deploy/Maintain
2. Agile

OBJECT ORIENTED PROGRAMMING

End of Part 2

And you ***SLACKERS*** still haven’t written a single line of code