

# CSC104 winter 2013

## Why and how of computing week 3

Danny Heap

heap@cs.toronto.edu

BA4270 (behind elevators)

<http://www.cdf.toronto.edu/~heap/104/W13/>

416-978-5899

Text: **Picturing Programs**

# how to solve it

it being a new problem

Clearly there's no fool-proof method, but there's some **techniques that often make progress**. It helps to write down the whole process:

- ▶ Understand the problem
- ▶ Devise (one or more) plan(s)
- ▶ Try the plan
- ▶ Look back

# paper folding?

try it out

- ▶ Understand the problem (what's given, what's required)?
- ▶ Devise a plan
- ▶ Try at least one plan (be ready to abandon it too)
- ▶ Look back

# In media res

racing with Alice

$A \leftrightarrow 1000001$   
Alice was beginning

Representing even simple information is hard. Let's race through this table:

Bits					Column	Row	0	1	2	3	4	5	6	7
b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	NUL	DLE	SP	0	@	P	.	p
0	0	0	1	1	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	0	1	0	STX	DC2	"	2	B	R	b	r
0	0	1	1	1	1	1	ETX	DC3	#	3	C	S	c	s
0	1	0	0	0	0	0	EOT	DC4	\$	4	D	T	d	t
0	1	0	1	0	1	0	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	0	1	0	ACK	SYN	&	6	F	V	f	v
0	1	1	1	1	1	1	BEL	ETB	'	7	G	W	g	w
1	0	0	0	0	0	0	BS	CAN	(	8	H	X	h	x
1	0	0	1	1	1	1	HT	EM	)	9	I	Y	i	y
1	0	1	0	0	1	0	LF	SUB	*	:	J	Z	j	z
1	0	1	1	1	1	1	VT	ESC	+	;	K	[	k	{
1	1	0	0	0	1	0	FF	FC	,	<	L	\	l	
1	1	0	1	1	1	1	CR	GS	-	=	M	]	m	}
1	1	1	0	1	0	1	SO	RS	.	>	N	^	n	~
1	1	1	1	1	1	1	SI	US	/	?	O	_	o	DEL



# Early devices

tally systems

*read/write many times → read only*



Clay tablets, read-only when baked,  
read/write when sundried,  
have been in use for at least 5,000 years.

*heavy to carry about*

Abacuses, or abaci, have been  
in use for nearly as long

*human moves  
beads about  
represent  
numbers.*



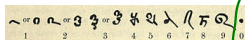
# Number systems and gears

ancient world

XV  
XIII .

15  
13

9  
90



Hindu-Arabic numbers: positional notation,  
and zero over 2000 years ago  
slick algorithms, e.g. long multiplication

*craft - ability to forge + machine  
metals*

Antikythera mechanism  
make us re-think  
ancient technical skills



*seems  
centuries  
ahead  
of its  
time*

# Gears and rules

machine age



*log of a numbers*

Add powers (logs) to multiply quickly, extract roots

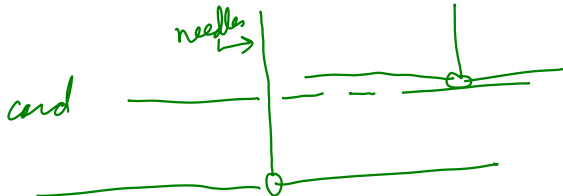
Read the gears to extract taxes — Pascaline



*units + 10s*

# Looms and engines

industrial revolution

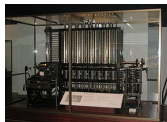


Jacquard loom combined steam and punch cards for automatic patterns

useful.

logs, trig

Babbages difference engine would have evaluated polynomials like  $3x^3 + 5x^2 - 7x + 9$



technical skill locking



# gears, pins, and electricity

digital and analog before tubes

used + financed for the  
1890 U.S. census -



data stored in punched cards  
manipulated by pins and  
electricity last for decades

analog computers model world  
using smoothly-varying quantities  
such as water



# programmable or electronic...

...but not both?



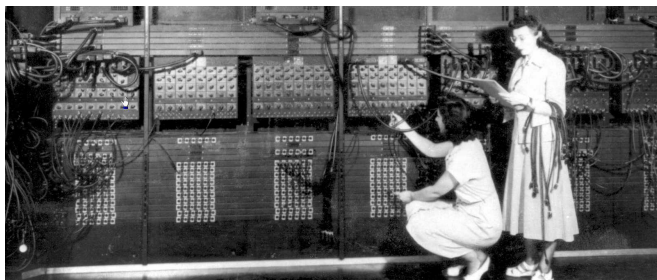
“programmable” (cards) but  
not electronic (relays)  
the Zuse Z1

electronic but not programmable  
dedicated to one calculation  
the Atanosoff-Berry



# when computers were women

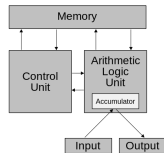
for a while



Eniac's first programmers were women  
known for clear-thinking, manual dexterity, and speed  
... human labour was cheaper than computer cycles  
dozens of cubic metres, programmed by connecting pins

# stored programs, faster switches

getting modern



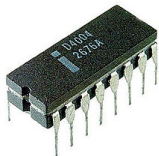
the same memory for data and programs is now the typical design

tubes were big, hot, slow  
compared to transistors  
... which just keep shrinking



# your (grand)parent's computer

smaller, faster ...



perhaps thanks to sputnik  
the computing power of eniac  
fits in your hand by 1970

mass-produced desktops  
landed with a clunk  
by 1980s



# Notes