Lecture 5

The Acceptance Problem for TMs

$$A_{TM} = \{ <M,w> \mid M \text{ is a TM } \& \ w \in L(M) \}$$

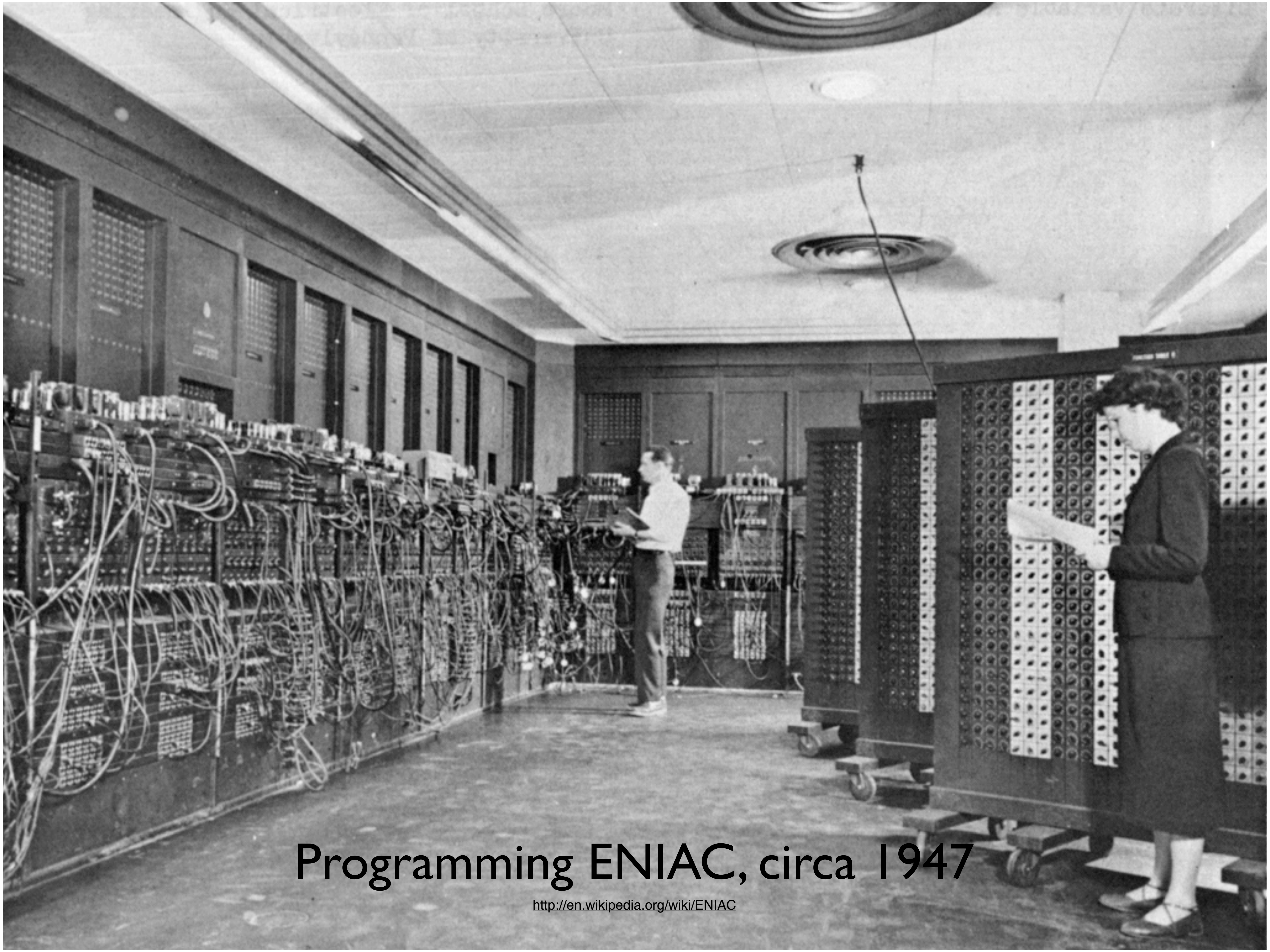
Theorem: A_{TM} is Turing recognizable

Pf: It is recognized by a TM U that, on input <M,w>, simulates M on w step by step. U accepts iff M does. □

U is called a *Universal Turing Machine*

(Ancestor of the stored-program computer)

Note that U is a recognizer, not a decider.



Programming ENIAC, circa 1947

http://en.wikipedia.org/wiki/ENIAC

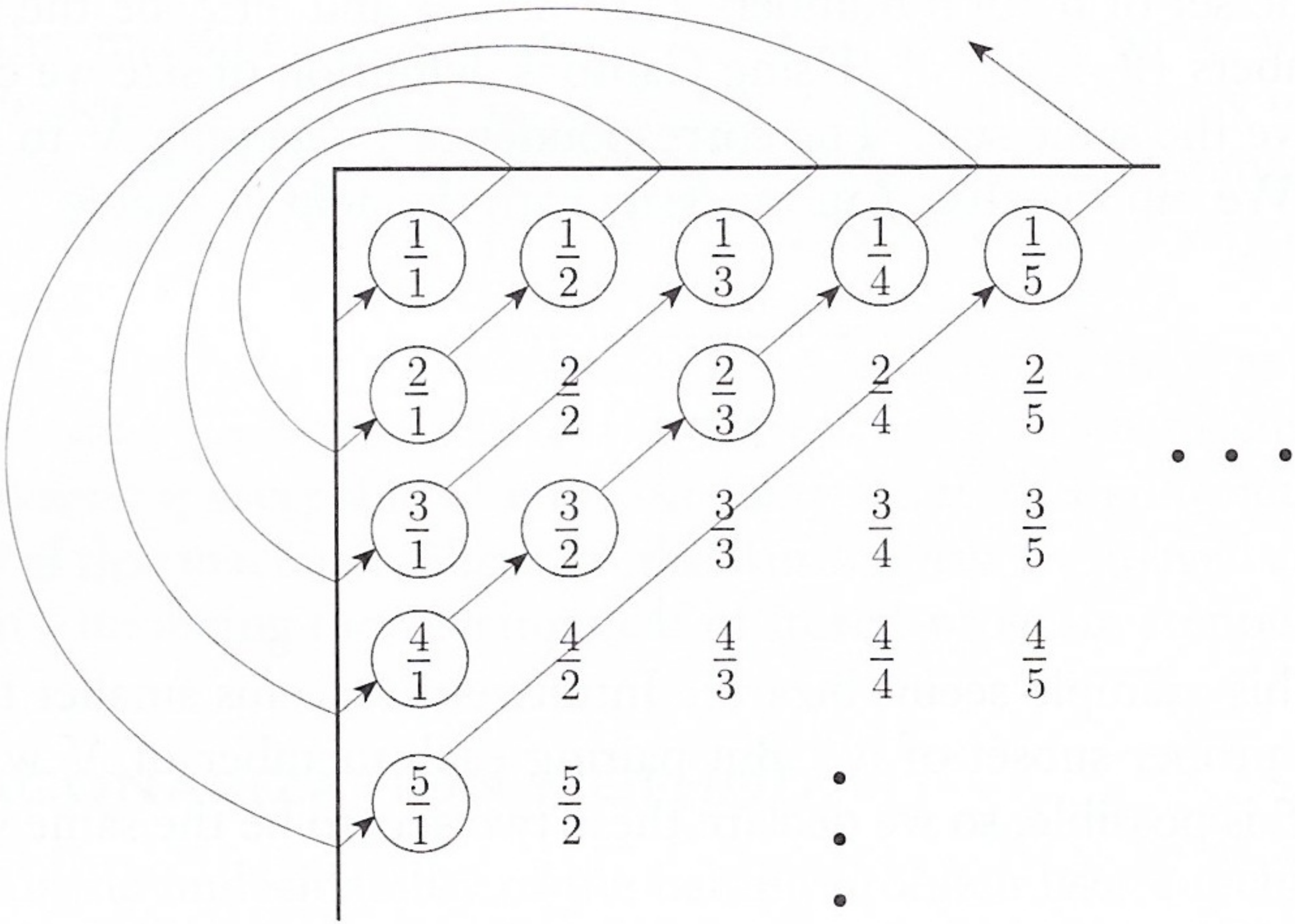
Cardinality

Two sets have equal cardinality if there is a bijection between them

A set is *countable* if it is finite or has the same cardinality as the natural numbers

Examples:

- Σ^* is countable (think of strings as base-$|\Sigma|$ numerals)
- Even natural numbers are countable: $f(n) = 2n$
- The Rationals are countable



$\frac{1}{1}$ $\frac{1}{2}$ $\frac{1}{3}$ $\frac{1}{4}$ $\frac{1}{5}$
$\frac{2}{1}$ $\frac{2}{2}$ $\frac{2}{3}$ $\frac{2}{4}$ $\frac{2}{5}$
$\frac{3}{1}$ $\frac{3}{2}$ $\frac{3}{3}$ $\frac{3}{4}$ $\frac{3}{5}$
$\frac{4}{1}$ $\frac{4}{2}$ $\frac{4}{3}$ $\frac{4}{4}$ $\frac{4}{5}$
$\frac{5}{1}$ $\frac{5}{2}$

More cardinality facts

If $f: A \to B$ in an injective function ("1-1", but not necessarily "onto"), then

$|A| \leq |B|$

(Intuitive: f is a bijection from A to its range, which is a subset of B, and B can't be smaller than a subset of itself.)

Theorem (Cantor-Schroeder-Bernstein):

If $|A| \leq |B|$ and $|B| \leq |A|$ then $|A| = |B|$

The Reals are Uncountable

Suppose they were

List them in order

Define X so that its i^{th} digit $\neq$ i^{th} digit of i^{th} real

Then X is *not in the list*

Contradiction

	int	1	2	3	3	5	
1	0.	0	0	0	0	0	
2	3.	1	4	1	5	9	
3	0.	3	3	3	3	3	
4	0.	5	0	0	0	0	...
5	2.	7	1	8	2	8	
6	41.	9	9	9	9	9	
				⋮			⋱

X	1.	2	4	1	3	8	...

A detail: avoid .000..., .9999... in X

Number of Languages in Σ^* is Uncountable

Suppose they were

List them in order

Define L so that $w_i \in L$ $\Leftrightarrow$ $w_i \notin L_i$

Then L is *not in the list*

Contradiction

	w_1	w_2	w_3	w_4	w_5	w_6	
L_1	**0**	0	0	0	0	0	
L_2	1	**1**	1	1	1	1	
L_3	0	1	**0**	1	0	1	
L_4	0	1	0	**0**	0	0	…
L_5	1	1	1	0	**0**	0	
L_6	1	1	1	1	0	**1**	
			⋮				⋱

L	1	0	1	1	1	0	…

“Most” languages are neither Turing recognizable nor Turing decidable

Pf:

“< >” maps TMs into Σ^*, a countable set, so the set of TMs, and hence of Turing recognizable languages is also countable; Turing decidable is a subset of Turing recognizable, so also countable. But by the previous result, the set of all languages is *un*countable.

A *specific* non-Turing-recognizable language

Let M_i be the TM encoded by w_i, i.e. $<M_i> = w_i$

(M_i = some default machine, if w_i is an illegal code.)

i, j entry tells whether M_i accepts w_j

Then D is *not recognized by any TM*

	w_1	w_2	w_3	w_4	w_5	w_6	
$<M_1>$	0	0	0	0	0	0	
$<M_2>$	1	1	1	1	1	1	
$<M_3>$	0	1	0	1	0	1	
$<M_4>$	0	1	0	0	0	0	…
$<M_5>$	1	1	1	0	0	0	
$<M_6>$	0	1	0	0	0	1	
			⋮			⋱	

D	1	0	1	1	1	0	...

Theorem: The class of Turing recognizable languages is *not* closed under complementation.

Proof:

The *complement* of D, *is* Turing recognizable:

On input w_i, run $<M_i>$ on w_i (= $<M_i>$); accept if it does. E.g. use a universal TM on input $<M_i,<M_i>>$

Theorem: The class of Turing decidable languages *is* closed under complementation.

Proof:

Flip q_{accept}, q_{reject}

Decidable $\subsetneq$ Recognizable