Lecture 6 - Turing Machines and Infinity

Eric A. Autry

Midterm is postponed.

Will now be **in class on Tuesday, October 2nd**.

Last time: CFGs, PDAs, Turing Machines

This time: Turing Machines and Infinity

Next time: Church-Turing Thesis

Homework 2 is due **today.**

Homework 3 will be posted on Sakai and due **next Thursday the 27th.**

Turing Machine Example 1

For Turing Machines, we typically describe an *algorithm* that the machine will follow for its computations.

Ex:

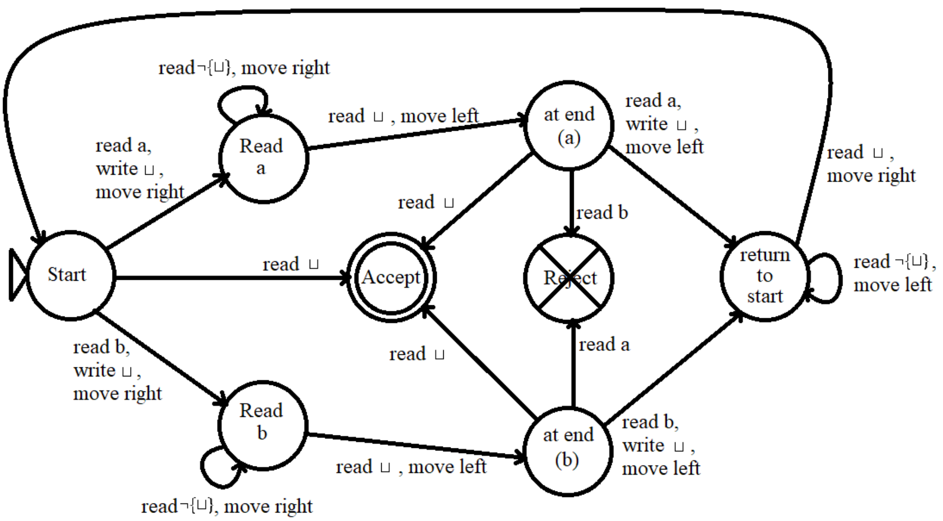
$$\{w \mid w \text{ is a palindrome}\}.$$

Let's work with the test input 'abba'.

1. Read the first letter and cross it off (replace it with an '$\sqcup$').
2. Move to the end of the string that is not yet crossed off.
   - If there was only one letter left, accept because a single letter is always a palindrome (or it was the unimportant middle letter).
3. Read the last letter and cross it off (replace it with an '$\sqcup$').
   - If the letters were not the same, halt and reject.
4. Move to the start of the tape that is not yet crossed off.
   - If there are no letters left, accept because the string had an even length and was a palindrome.
5. Go to step 1.

Turing Machine Example 1

Let’s build the actual machine...



Formal Definition of a Turing Machine

A Turing Machine is a 7-tuple, $(Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$, where Q, Σ, and Γ are all finite sets, and

1. Q is the set of states,
2. Σ is the input alphabet not containing the blank symbol $\sqcup$,
3. Γ is the tape alphabet, where $\sqcup \in \Gamma$ and $\Sigma \subset \Gamma$,
4. $\delta : Q \times \Gamma \to Q \times \Gamma \times \{left, right\}$ is the transition function,
   - Given a state and a symbol read from the tape, return a new state, possibly a new symbol to write, and possibly a movement to make.
5. $q_0 \in Q$ is the start state,
6. $q_{accept} \in Q$ is the accept state,
7. $q_{reject} \in Q$ is the reject state, where $q_{reject} \neq q_{accept}$.

Turing Machine Example 2

$$\{w\#w \mid w \in \{0,1\}^*\}$$

i.e., we are checking if the word before the $\#$ is equal to the word after the $\#$.

1. Check the first letter and cross it off.
2. Move to the first letter after the $\#$ that is not crossed off.
    - If there are none left, halt and reject.
3. Check that letter and cross it off.
    - If they were not the same, halt and reject.
4. Return to the first letter of the string before the $\#$ that is not crossed off.
5. If that string has no letters left, check if the string after the $\#$ has letters left.
    - If the second string has letters left, halt and reject.
    - If both strings are empty, halt and accept.
    - Else, return to step 1.

Turing Machine Example 3

$$\{\#w_1\#w_2\ldots\#w_k \mid \text{each } w_i \in \{0,1\}^* \text{ and } w_i \neq w_j \text{ for all } i \neq j\}$$

This is the element distinctness problem, which only accepts if the words are all different.

1. Mark the first $\#$ as $\dot{\#}$
   - If no first $\#$, empty string so accept.
   - If not a $\#$, reject.
2. Scan right and mark the second $\#$ as $\dot{\#}$
   - If no second $\#$, there was one string left so accept.
3. Compare the words to the right of the marked $\dot{\#}$.
   - If they are the same, reject.
4. Move the second mark to the next $\#$ and go to step 3.
   - If there are no $\#$ remaining, move the first mark to the next $\#$ and go to step 2.

Important note from this example: we can mark symbols without deleting them. We can use different kinds of marks together if we want: $\dot{\#}$, $\hat{\#}$, $\bar{\#}$, etc.

Decidable Languages

If a Turing machine halts on all inputs and either accepts or rejects, the language it recognizes is called a **decidable** language. (These are also known as recursive languages.)

When this happens, we say that the Turing machine **decides** the language.

What if there is an input that causes the Turing Machine to never halt?

Semidecidable Languages

If a Turing machine

- halts and accepts all strings in a language A,

and for strings that are not in A, **either**:

- halts and rejects, **or**
- loops forever,

we say the Turing machine **recognizes** the language A and call the language **semidecidable.** (These are also known as Turing-recognizable or recursively enumerable languages.)

Note: for these semidecidable languages, we are treating infinite loops as a form of rejection.

Turing Machine Example 4

Can we decide if an undirected graph G is a connected?

1. Select the first vertex of G and mark it.
2. Repeat until no new vertices are marked by the machine:
   - For each vertex in G, mark it if it is connected by an edge to a vertex that is already marked.
3. Scan the vertices of G:
   - It they are all marked, accept.
   - Else, reject.

Wait... how is graph G stored on the tape?

Turing Machine Example 4

Wait... how is graph G stored on the tape?

We can label the vertices in a graph (typically by numbering them), and can use these labels to create a mathematical representation of the graph:

- If a graph G contains vertices i and j, the pair (i, j) represents an edge connecting the two.
- In an undirected graph, the order of the pair does not matter, with (i, j) and (j, i) representing the same edge.
- In a directed path, the order of the pair matters: (i, j) is the edge pointing from i to j, while (j, i) is the edge pointing from j to i.
- If V is the set of vertices in a graph, and E is the set of edges, we say $G = (V, E)$.

$$G = (\ \{1, 2, 3, 4\},\ \{\,(1, 2),\ (1, 3),\ (1, 4),\ (2, 3),\ (2, 4),\ (3, 4)\,\}\ ).$$

Other Machines as Inputs

To take the graph G as an input string, we write

$$\{\langle G \rangle \mid G \text{ is a connected undirected graph}\}$$

Note: all of the machines we have talked about (DFAs, NFAs, Regular Expressions, CFGs, PDAs, TMs) each have mathematical descriptions.

This means that we can encode them all as strings and input these machines into other Turing Machines.

Ex: the acceptance problem for DFAs.

$$A_{DFA} = \{\langle B, w \rangle \mid B \text{ is a DFA that accepts input string } w\}$$

Acceptance Problem for DFAs

$$A_{DFA} = \{\langle B, w\rangle \mid B \text{ is a DFA that accepts input string } w\}$$

1. Simulate machine B on input w.
   - (a) Mark the start state and the first input of w.
   - (b) Read the marked input of w and the marked state of B.
   - (c) Follow the corresponding transition, marking the new state of B and the next symbol of w.
   - (d) If the next symbol is not black, return to Step 1(b), otherwise continue to Step 2.
2. If B ends in an accepting state, accept. If it ends in a nonaccepting state, reject.

Note: we can do this for NFAs, Regular Expressions, PDAs, and CFGs too!

Context Free vs Decidable vs Semidecidable

Note: we can do this for NFAs, Regular Expressions, PDAs, and CFGs too!

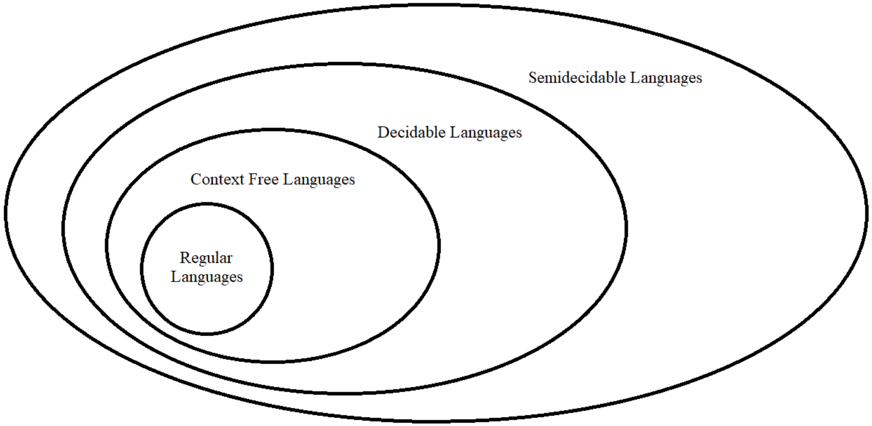
What is so important about that?

It means that every context-free language is decidable.

But, remember palindromes: not every decidable language is context-free.

Also, note that every decidable language is semidecidable (since decidable means halts and accepts or rejects, while semidecidable allows for looping).

Context Free vs Decidable vs Semidecidable



Question: are all languages decidable?

Correspondence

Consider a function $f : A \to B$.

A function is called **one-to-one** if it never maps two different inputs to the same output.

- i.e., if $a_1 \neq a_2$, then $f(a_1) \neq f(a_2)$.

A function is called **onto** if it hits every element in B.

- i.e., for every $b \in B$ there exists an $a \in A$ such that $f(a) = b$.

We say that the sets A and B are the **same size** if there exists a function $f : A \to B$ that is one-to-one and onto. We call this function a **correspondence**.

Even Natural Numbers

- Let $\mathbb{N}$ be the set of natural numbers: $\{1, 2, 3, \ldots\}$.
- Let $\mathbb{E}$ be the set of even natural numbers: $\{2, 4, 6 \ldots\}$.

Which set is larger?

Theorem
The sets $\mathbb{N}$ and $\mathbb{E}$ are the same size.

Proof: Simply define the function

$$f(n) = 2n.$$

We can visualize this as:

n	$f(n)$
1	2
2	4
3	6
.	.
.	.
.	.

Countable Sets

A set is **countable** if it either is finite or has the same size as $\mathbb{N}$.

This means that a set is countable if we can assign the natural numbers to the elements of the set.

i.e., if we can **count** the elements of the set.

Rational Numbers

- Consider the positive rational numbers $\left\{ \frac{m}{n} \,\middle|\, m, n \in \mathbb{N} \right\}$.

Theorem
The positive rational numbers are countable.

Proof:

What about the real numbers?

Rational Numbers

Theorem

The positive rational numbers $\mathbb{Q}$ are countable.

Proof: Let's write the positive rationals in the following table, with the numerators down the columns and the denominators across the rows:

	1	2	3	4	5	...
1	1/1	1/2	1/3	1/4	1/5	...
2	2/1	2/2	2/3	2/4	2/5	...
3	3/1	3/2	3/3	3/4	3/5	...
4	4/1	4/2	4/3	4/4	4/5	...
5	5/1	5/2	5/3	5/4	5/5	...
.	.	.	.	.	.	
.	.	.	.	.	.	
.	.	.	.	.	.	

Rational Numbers

Theorem

The positive rational numbers $\mathbb{Q}$ *are countable.*

Proof: Now let's cross out any repeated rationals:

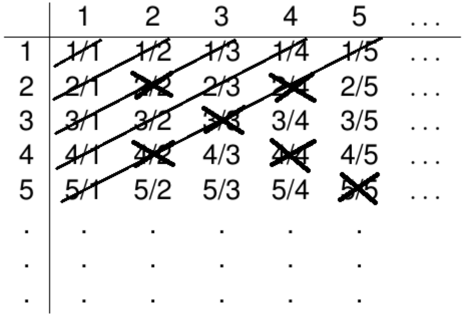
	1	2	3	4	5	...
1	1/1	1/2	1/3	1/4	1/5	...
2	2/1	~~2/2~~	2/3	~~2/4~~	2/5	...
3	3/1	3/2	~~3/3~~	3/4	3/5	...
4	4/1	~~4/2~~	4/3	~~4/4~~	4/5	...
5	5/1	5/2	5/3	5/4	~~5/5~~	...
.	.	.	.	.	.	
.	.	.	.	.	.	
.	.	.	.	.	.	

Rational Numbers

Theorem

The positive rational numbers $\mathbb{Q}$ *are countable.*

Proof: Consider the diagonal lines. We will follow each diagonal and start numbering as we encounter new rationals:



	1	2	3	4	5	...
1	1/1	1/2	1/3	1/4	1/5	...
2	2/1	$\xcancel{2/2}$	2/3	$\xcancel{2/4}$	2/5	...
3	3/1	3/2	$\xcancel{3/3}$	3/4	3/5	...
4	4/1	$\xcancel{4/2}$	4/3	$\xcancel{4/4}$	4/5	...
5	5/1	5/2	5/3	5/4	$\xcancel{5/5}$	...
⋮	⋮	⋮	⋮	⋮	⋮	

Rational Numbers

Theorem

The positive rational numbers $\mathbb{Q}$ are countable.

Proof: So, the order of the rationals we have devised is

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

Interesting note: each diagonal was a finite set of rationals, and there were a countable number of diagonals.

Why? We can consider the 1st diagonal, the 2nd diagonal, the 3rd diagonal, etc... (i.e., we can count the diagonals).

So we can now conclude that the combined set of elements from a countable number of finite sets is itself a countable set.

Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:

Originally proved by Georg Cantor in 1874 at the age of 29.

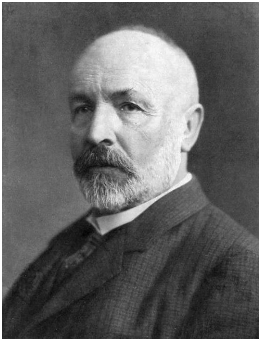
Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:

Diagonalization proof developed by Cantor in 1891.



Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:

We can write down decimal representations for all reals.

$$\begin{array}{cccccccc}
0 & . & 5 & 0 & 0 & 0 & 0 & \ldots \\
3 & . & 1 & 4 & 1 & 5 & 9 & \ldots \\
2 & . & 7 & 1 & 8 & 2 & 8 & \ldots \\
1 & . & 4 & 1 & 4 & 2 & 1 & \ldots \\
1 & . & 7 & 3 & 2 & 0 & 5 & \ldots \\
. & . & . & . & . & . & . & \\
. & . & . & . & . & . & . & \\
. & . & . & . & . & . & . &
\end{array}$$

Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:
By way of contradiction, assume that $\mathbb{R}$ is countable.

0	.	5	0	0	0	0	...
3	.	1	4	1	5	9	...
2	.	7	1	8	2	8	...
1	.	4	1	4	2	1	...
1	.	7	3	2	0	5	...
.	.	.	.	.	.	.	
.	.	.	.	.	.	.	
.	.	.	.	.	.	.	

Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:
Then we can put them in a numbered list like so:

1	0	.	5	0	0	0	0	...
2	3	.	1	4	1	5	9	...
3	2	.	7	1	8	2	8	...
4	1	.	4	1	4	2	1	...
5	1	.	7	3	2	0	5	...
.	.	.	.	.	.	.	.	
.	.	.	.	.	.	.	.	
.	.	.	.	.	.	.	.	

Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:

Let's find a number between 0 and 1 that is not in that list.

1	0	.	5	0	0	0	0	...	0	.
2	3	.	1	4	1	5	9	...		
3	2	.	7	1	8	2	8	...		
4	1	.	4	1	4	2	1	...		
5	1	.	7	3	2	0	5	...		
.	.	.	.	.	.	.	.			
.	.	.	.	.	.	.	.			
.	.	.	.	.	.	.	.			

Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:

Look at the 1st digit of the 1st number and change it.

$$
\begin{array}{c|cccccccccccc}
1 & 0 & . & \textcircled{5} & 0 & 0 & 0 & 0 & \dots & & 0 & . & 4 \\
2 & 3 & . & 1 & 4 & 1 & 5 & 9 & \dots & & & & \\
3 & 2 & . & 7 & 1 & 8 & 2 & 8 & \dots & & & & \\
4 & 1 & . & 4 & 1 & 4 & 2 & 1 & \dots & & & & \\
5 & 1 & . & 7 & 3 & 2 & 0 & 5 & \dots & & & & \\
. & . & . & . & . & . & . & . & & & & & \\
. & . & . & . & . & . & . & . & & & & & \\
. & . & . & . & . & . & . & . & & & & &
\end{array}
$$

Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:

Look at the 2nd digit of the 2nd number and change it.

$$
\begin{array}{c|cccccccc}
1 & 0 & . & \textcircled{5} & 0 & 0 & 0 & 0 & \ldots \\
2 & 3 & . & 1 & \textcircled{4} & 1 & 5 & 9 & \ldots \\
3 & 2 & . & 7 & 1 & 8 & 2 & 8 & \ldots \\
4 & 1 & . & 4 & 1 & 4 & 2 & 1 & \ldots \\
5 & 1 & . & 7 & 3 & 2 & 0 & 5 & \ldots \\
. & . & . & . & . & . & . & . & \\
. & . & . & . & . & . & . & . & \\
. & . & . & . & . & . & . & . &
\end{array}
\qquad
0\ .\ 4\ 7
$$

Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:

Look at the 3rd digit of the 3rd number and change it.

$$\begin{array}{c|cccccccc}
1 & 0 & . & \textcircled{5} & 0 & 0 & 0 & 0 & \dots \\
2 & 3 & . & 1 & \textcircled{4} & 1 & 5 & 9 & \dots \\
3 & 2 & . & 7 & 1 & \textcircled{8} & 2 & 8 & \dots \\
4 & 1 & . & 4 & 1 & 4 & 2 & 1 & \dots \\
5 & 1 & . & 7 & 3 & 2 & 0 & 5 & \dots \\
. & . & . & . & . & . & . & . & \\
. & . & . & . & . & . & . & . & \\
. & . & . & . & . & . & . & . &
\end{array}
\qquad\qquad
0 \;.\; 4 \quad 7 \quad 6$$

Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:

Look at the 4th digit of the 4th number and change it.

$$\begin{array}{c|cccccccc}
1 & 0 & . & \textcircled{5} & 0 & 0 & 0 & 0 & \dots \\
2 & 3 & . & 1 & \textcircled{4} & 1 & 5 & 9 & \dots \\
3 & 2 & . & 7 & 1 & \textcircled{8} & 2 & 8 & \dots \\
4 & 1 & . & 4 & 1 & 4 & \textcircled{2} & 1 & \dots \\
5 & 1 & . & 7 & 3 & 2 & 0 & 5 & \dots \\
. & . & . & . & . & . & . & . & \\
. & . & . & . & . & . & . & . & \\
. & . & . & . & . & . & . & . &
\end{array}
\qquad\qquad
0\ .\ 4\ \ 7\ \ 6\ \ 1$$

Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:
Look at the 5th digit of the 5th number and change it. etc...

1	0	.	$\textcircled{5}$	0	0	0	0	. . .
2	3	.	1	$\textcircled{4}$	1	5	9	. . .
3	2	.	7	1	$\textcircled{8}$	2	8	. . .
4	1	.	4	1	4	$\textcircled{2}$	1	. . .
5	1	.	7	3	2	0	$\textcircled{5}$	. . .
.	.	.	.	.	.	.	.	
.	.	.	.	.	.	.	.	
.	.	.	.	.	.	.	.	

0 . 4 7 6 1 9

Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:

This new number cannot be in the original list.

$$
\begin{array}{c|cccccccc}
1 & 0 & . & \textcircled{5} & 0 & 0 & 0 & 0 & \ldots \\
2 & 3 & . & 1 & \textcircled{4} & 1 & 5 & 9 & \ldots \\
3 & 2 & . & 7 & 1 & \textcircled{8} & 2 & 8 & \ldots \\
4 & 1 & . & 4 & 1 & 4 & \textcircled{2} & 1 & \ldots \\
5 & 1 & . & 7 & 3 & 2 & 0 & \textcircled{5} & \ldots \\
. & . & . & . & . & . & . & . & \\
. & . & . & . & . & . & . & . & \\
. & . & . & . & . & . & . & . &
\end{array}
\qquad
0\ .\ 4\ \ 7\ \ 6\ \ 1\ \ 9
$$

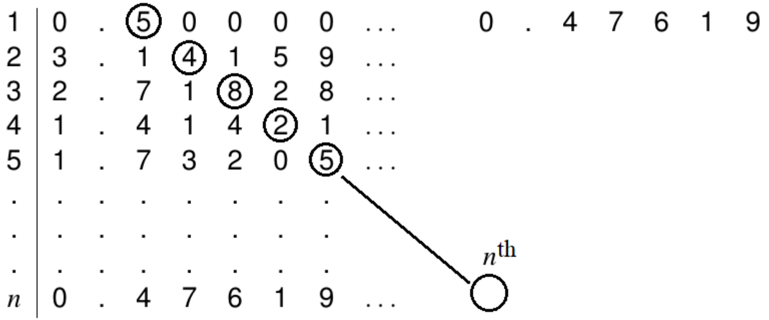
Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:

If it was the nth number, its nth digit would be changed.



Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:

What if we add this new number to our list?

$$\begin{array}{c|cccccccc}
1 & 0 & . & 5 & 0 & 0 & 0 & 0 & \ldots \\
2 & 3 & . & 1 & 4 & 1 & 5 & 9 & \ldots \\
3 & 2 & . & 7 & 1 & 8 & 2 & 8 & \ldots \\
4 & 1 & . & 4 & 1 & 4 & 2 & 1 & \ldots \\
5 & 1 & . & 7 & 3 & 2 & 0 & 5 & \ldots \\
. & . & . & . & . & . & . & . & \\
. & . & . & . & . & . & . & . & \\
. & . & . & . & . & . & . & . & \\
n & 0 & . & 4 & 7 & 6 & 1 & 9 & \ldots
\end{array}$$

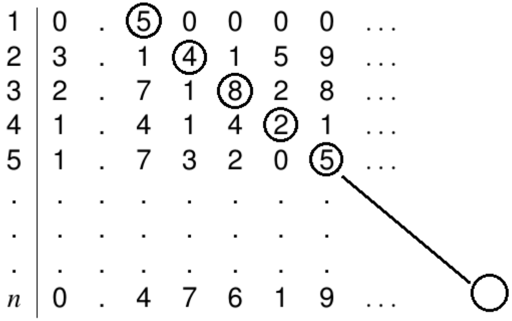
Real Numbers

Theorem

The real numbers $\mathbb{R}$ are uncountable.

Proof:

Still incomplete because we can just repeat the process.



$$
\begin{array}{c|cccccccc}
1 & 0 & . & \textcircled{5} & 0 & 0 & 0 & 0 & \dots \\
2 & 3 & . & 1 & \textcircled{4} & 1 & 5 & 9 & \dots \\
3 & 2 & . & 7 & 1 & \textcircled{8} & 2 & 8 & \dots \\
4 & 1 & . & 4 & 1 & 4 & \textcircled{2} & 1 & \dots \\
5 & 1 & . & 7 & 3 & 2 & 0 & \textcircled{5} & \dots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\
n & 0 & . & 4 & 7 & 6 & 1 & 9 & \dots \quad \bigcirc
\end{array}
$$