# Handout 9a - Countability, Turing Reductions, Proving Undecidability

#### Owen Keith and Owen Terry

### COMS 3261 Fall 2024

## 1 Countability

**Definition 1.** A set S is countable if S is finite or  $|S| = |\mathbb{N}|$ . We say S is uncountable otherwise. Examples of countable sets include the set of all integers,  $\mathbb{Z}$ , and the set of all rational numbers,  $\mathbb{Q}$ .

**Definition 2.** An equivalent definition for countability is, a set S is countable if and only if each element of S can be written as a unique finite string.

**Definition 3.** Recall that two sets S and T have the same cardinality, denoted as |S| = |T|, if there exists a bijection  $f: S \to T$ . If no such bijection exists, then we say that S and T have different cardinalities and write  $|S| \neq |T|$ .

**Definition 4.** Recall the definitions of injective, surjective, and bijective functions from the Discrete Math Handout: Let S and T be sets and  $f: S \to T$  be a function. Then we say that f is injective (or one-to-one), or an injection, if for any  $x, y \in S$ , f(x) = f(y) implies that x = y. We say that f is surjective (or onto), or a surjection, if for all  $y \in T$ , there exists some  $x \in S$  such that f(x) = y. We say that f is bijective, or a bijection, if f is both injective.

Given these definitions, the above discussion translates to saying that a set S has n elements if and only if there exists a bijection  $f : \{1, 2, ..., n\} \to S$ .

**Example 1.** Let  $\mathbb{Z}$  be the set of all integers, and let  $\mathbb{N} = 0, 1, 2, 3, ...$  be the set of all non-negative integers. Then  $|\mathbb{N}| = |\mathbb{Z}|$ : consider the function  $f : \mathbb{N} \to \mathbb{Z}$  given by:

$$f(n) = \begin{cases} n/2 & \text{if n is even} \\ -(n-1)/2 & \text{if n is odd} \end{cases}$$

**Theorem 1.** Any countable union of countable sets is countable, i.e. if  $\{E_i\}_{n=1}^{\infty}$  are countable sets, then

$$S = \bigcup_{n=1}^{\infty} E_n$$

is countable.

**Theorem 2.** Any finite product of countable sets is countable, i.e. if  $E_1, ..., E_n$  are countable sets, then

$$E_1 \times ... \times E_n = \{(a_1, ..., a_n) \mid a_i \in E_i\}$$

is countable.

**Theorem 3.** If L is a countable set, then every subset of L is countable, i.e., every set S where  $S \subseteq L$  is countable.

**Example 2.** Let  $\Sigma$  be a finite alphabet, then  $\Sigma^*$  is countable.

*Proof.* Notice that for every  $n \in \mathbb{N}$  that  $\Sigma^n$  is finite; for any string  $s \in \Sigma^n$  there are  $|\Sigma|$  choices for each character of s, so  $|\Sigma^n| = |\Sigma|^n$  is finite. Now recall that

$$\Sigma^* = \bigcup_{n=0}^{\infty} \Sigma^n$$

is a countable union of finite sets, so  $\Sigma^*$  is countable as a result.

Some more examples of sets we saw in class that are countable:

- { $\langle M \rangle \mid M$  is a TM} is countable, since each TM encoding  $\langle M \rangle$  can be written as a unique finite string.
- $\{L \mid L \text{ is recognizable}\}$  and  $\{L \mid L \text{ is decidable}\}$  are countable.
- Every subset  $L \subseteq \Sigma^*$  is countable.

Here are some examples of uncountable sets:

- $\mathbb{R}$ , the set of all real numbers is uncountable. We saw in class the diagonalization argument for proving this.
- The set of all languages, i.e., the set of all subsets of  $\Sigma^*$  is uncountable.
- The power set of the natural numbers,  $\mathbb{P}(\mathbb{N})$ , which is the set of all subsets of  $\mathbb{N}$ , is uncountable. In general, for a set S, if |S| is non-finite, then  $\mathbb{P}(S)$  is uncountable.

# 2 Turing Reductions and Undecidability

**Definition 5.** We say a language A is Turing-reducible to a language B, written  $A \leq_T B$ , if given an oracle that decides B, there exists a decider for A.

We use the  $\leq$  sign here because, intuitively, A is "easier" than (or equal to) B. Essentially,  $A \leq_T B$  means that if B is decidable, so is A. This gives us the following theorem:

**Theorem 4.** If  $A \leq_T B$  and B is decidable, then A is decidable.

and taking the contrapositive gives us the following (very useful) corollary:

**Corollary 1.** If  $A \leq_T B$  and A is undecidable, then B is undecidable.

Thus, if we know that a language is undecidable, we can use this to show many other languages are undecidable!

#### 2.1 Examples of undecidable languages

- 1.  $A_{TM} := \{ \langle M, w \rangle \mid M \text{ is a TM that accepts } w \}$
- 2.  $E_{TM} := \{ \langle M \rangle \mid M \text{ is a TM with } L(M) \neq \emptyset \}$
- 3.  $HALT_{TM} := \{ \langle M, w \rangle \mid M \text{ is a TM that halts on } w \}$
- 4.  $EQ_{TM} := \{ \langle M_1, M_2 \rangle \mid M_1, M_2 \text{ are TMs and } L(M_1) = L(M_2) \}$

#### 2.2 Exercises

- 1. Prove that  $HALT_{TM} \leq_T A_{TM}$ .
- 2. Prove that  $L = \{ \langle M, D \rangle \mid M \text{ is a TM}, D \text{ is a DFA}, \text{ and } L(M) = L(D) \}$  is undecidable.
- 3. Prove that the following are equivalent:  $A \leq_T B, \overline{A} \leq_T B, A \leq_T \overline{B}, \overline{A} \leq_T \overline{B}$ .

# 3 Using Rice's theorem to prove undecidability

Note that many of the undecidable languages we have learned about fit a common pattern. That is, they are languages of the form  $\{\langle M \rangle \mid M \text{ is a TM and } L(M) \text{ satisfies...}\}$ . We call such languages "properties of recognizable languages." Formally,

**Definition 6.** *P* is a property of recognizable languages if  $P \subset \{\langle M \rangle \mid M \text{ is a TM}, \text{ and if } L(M_1) = L(M_2) \text{ then } \langle M_1 \rangle \in P \iff \langle M_2 \rangle \in P \}.$ 

In fact, there is a very convenient way to tell if a language of this form is decidable, using the following theorem:

**Theorem 5.** Rice's twheorem: Let P be a non-trivial property of recognizable languages. That is, P is a property of recognizable languages such that  $P \neq \emptyset$  and  $P \neq \{\langle M \rangle \mid M \text{ is a TM}\}$ . Then P is not decidable.

We can prove the theorem by showing that  $A_{TM}$  Turing-reduces to any non-trivial P. Since  $A_{TM}$  is undecidable, P cannot be decidable.

Since Rice's theorem seems to apply to a very broad class of languages, it is worth pointing out some types of languages to which it does **not** apply:

- 1. Trivial languages,  $L = \emptyset$  or  $L = \{ \langle M \rangle \mid M \text{ is a TM} \}$
- 2. Languages (decision problems) where the elements (inputs) are not encodings of TMs  $\langle M \rangle$ . This is because such languages do not represent TM properties.
- 3. Languages for which the TM property depends on the implementation (for example,  $\{\langle M \rangle \mid M \text{ is a TM that always moves right}\}$ ).

### 3.1 Exercises

Does Rice's theorem apply to the following languages? If not, determine whether or not the language is decidable with another method.

- 1.  $L = \{ \langle M \rangle \mid M \text{ is a TM and } M \text{ accepts } 0 \}$
- 2.  $L = \{ \langle M \rangle \mid M \text{ is a TM and } M \text{ has exactly two states} \}$
- 3.  $L = \{ \langle M \rangle \mid M \text{ is a TM and } M \text{ rejects } 0 \}$
- 4.  $E_{TM} = \{ \langle M \rangle \mid M \text{ is a TM and } L(M) = \emptyset \}$
- 5.  $L = \{ \langle M \rangle \mid M \text{ is a TM and } L(M) = \overline{A_{TM}} \}$
- 6.  $L = \{ \langle M \rangle \mid M \text{ is a TM and } L(M) \text{ is recognizable} \}$
- 7.  $L = \{ \langle M \rangle \mid M \text{ is a TM and } L(M) \text{ is decidable} \}$