COMP170 Spring 2017 Recitation 6

For each of the following sets, give the smallest complexity class in which it is contained. Classes are listed from smallest to largest. Write your choice A through D under each numbered language. Your choices are:

A. DECIDABLE
B. UNDECIDABLE but RECOGNIZABLE
C. UNDECIDABLE and UNRECOGNIZABLE but its compliment is RECOGNIZABLE
D. UNDECIDABLE and both it and its compliment are UNRECOGNIZABLE

Here, we include answers and informal, intuitive explanations. Proofs are below.

I: \{⟨M_1, M_2, M_3⟩ | L(M_1) ∩ L(M_2) = L(M_3)\}
(D) Consider that we can build a machine \(M'\) that accepts \(L(M_1) \cap L(M_2)\). Given \(M'\), the problem becomes identical to \(EQ_{TM}\), which we know is co-unrecognizable. So this language is also co-unrecognizable.

II: \{⟨M_1, M_2, M_3⟩ | |L(M_1)|, |L(M_2)|, and |L(M_3)| are finite and |L(M_1)| + |L(M_2)| < |L(M_3)|\}
(D) Determining even relative sizes of languages seems to require checking the machines’ behavior over all inputs. Since we need to determine relative sizes to verify that \(⟨M_1, M_2, M_3⟩\) is in, or out, of the language, both tasks seem impossible.

III: \{⟨M⟩ | M does not accept a string beginning with 1\}
(We assume ‘a string’ means ‘any string’. If ‘a’ means ‘some’, then this language is the complement of \(V\), and is type (D).)
(C) Consider the compliment of this language: \(M\) accepts some string beginning with 1. Given the similarity of the compliment to \(A_{TM}\), the compliment seems recognizable and undecidable. Since the compliment is recognizable and undecidable, this language is unrecognizable.

IV: \{⟨M, w, t⟩ | M accepts \(w\) within \(t\) steps\}
(A) In a finite amount of time, we can run \(M\) on input \(w\) for \(t\) steps and see if it accepts or not. There is no danger of looping because \(M\)’s computation is bounded.

V: \{⟨M⟩ | M accepts all strings beginning with 1\}
(D) In order to verify that \(M\) is in the language, we would have to check its behavior over all strings. In order to verify that \(M\) is out of the language, we would have to determine that there is some string that it does not accept, which we cannot do when \(M\) is looping.

VI: \(L(M)\) for some Turing machine \(M\)
(B) For any input \(x\), we can verify that it is in the language by running \(M\) on \(x\) and waiting for it to accept. However, in order to verify that some input \(y\) is not in the language, we would have to know that \(M\) does not accept on \(x\), which is not possible if \(M\) is looping.

VII: \{⟨M, w, q⟩ | M enters a state \(q\) during its computation of \(w\)\}
(B) We can verify that \(M\) is in the language by running it on \(w\) and waiting for it to enter state \(q\). However, to verify that \(M\) is not in the language, we would have to know that it never enters state \(q\), even when its computation might be infinitely long.
I: \( L = \{ \langle M_1, M_2, M_3 \rangle \mid L(M_1) \cap L(M_2) = L(M_3) \} \)

Solution: (D) UNDECIDABLE and both it and its compliment are UNRECOGNIZABLE.

We reduce \( L \) from \( A_{TM} \) and \( \overline{A_{TM}} \).

- \( A_{TM} \leq_m L \).
  Define the reduction function \( f \)
  \[
  f(\langle M, w \rangle) = \langle M_A, M_B, M_B \rangle
  \]
  where \( M_A \) and \( M_B \) are defined as follows
  \[
  \begin{align*}
  M_A \text{ on } x & \\
  1. \text{ If } x = 1, \text{ run } M & \text{ on } w \\
  & \text{if } M \text{ accepts, ACCEPT} \\
  & \text{if } M \text{ rejects, loop}
  \end{align*}
  \begin{align*}
  M_B \text{ on } x & \\
  1. \text{ If } x = 1, \text{ ACCEPT} \\
  2. \text{ Else, REJECT}
  \end{align*}
  \]
  Since \( M_A \) and \( M_B \) are finite and each of their instructions is computable, \( f \) is computable.
  Suppose \( \langle M, w \rangle \in A_{TM} \). Then \( M \) accepts \( w \), and \( M_A \) accepts ‘1’ and rejects all other inputs.
  So \( L(M_A) = L(M_B) = \{1\} \). Since \( \{1\} \cap \{1\} = \{1\} \), \( \langle M_A, M_B, M_B \rangle \in L \).
  Suppose \( \langle M, w \rangle \notin A_{TM} \). Then \( M \) either rejects or loops on \( w \). In either case, \( M_A \) loops on ‘1’ and rejects all other inputs.
  So \( L(M_A) = \phi \) and \( L(M_B) = \{1\} \). Since \( \phi \cap \{1\} = \phi \) and \( \phi \notin \{1\} \), \( \langle M_A, M_B, M_B \rangle \notin L \).

- \( \overline{A_{TM}} \leq_m L \).
  Define the reduction function \( g \)
  \[
  g(\langle M, w \rangle) = \langle M_A, M_B, M_C \rangle
  \]
  where \( M_A \) and \( M_B \) are defined as above, and \( M_C \) is defined as follows
  \[
  \begin{align*}
  M_C \text{ on } x & \\
  1. \text{ REJECT}
  \end{align*}
  \]
  Since \( M_A \), \( M_B \), and \( M_C \) are finite and each of their instructions is computable, \( g \) is a computable function.
  Suppose \( \langle M, w \rangle \in \overline{A_{TM}} \). Then \( M \) either rejects or loops on \( w \). So, \( L(M_C) = \phi \) and, as above, \( L(M_A) = \phi \) and \( L(M_B) = \{1\} \). Since \( \phi \cap \{1\} = \phi \), \( \langle M_A, M_B, M_C \rangle \in L \).
  Suppose \( \langle M, w \rangle \notin \overline{A_{TM}} \). Then \( \langle M, w \rangle \in A_{TM} \) and \( M \) accepts on \( w \). So, \( L(M_C) = \phi \) and, as above, \( L(M_A) = \{1\} \) and \( L(M_B) = \{1\} \). Since \( \{1\} \cap \{1\} = \{1\} \notin \phi \), \( \langle M_A, M_B, M_C \rangle \notin L \).

Therefore, \( f \) is a reduction from \( A_{TM} \) to \( L \) and \( g \) is a reduction from \( \overline{A_{TM}} \) to \( L \). So \( L \) is co-unrecognizable.
II: $L = \{ \langle M_1, M_2, M_3 \rangle \mid |L(M_1)|, |L(M_2)|, \text{ and } |L(M_3)| \text{ are finite and } |L(M_1)| + |L(M_2)| < |L(M_3)| \}$

Solution: (D) UNDECIDABLE and both it and its compliment are UNRECOGNIZABLE

We reduce $L$ from $A_{TM}$ and $\overline{A_{TM}}$.

• $A_{TM} \leq_m L$.

Define the reduction function $f$

$$f(\langle M, w \rangle) = \langle M_A, M_A, M_B \rangle$$

where $M_A$ and $M_B$ are defined as follows

$M_A$ on $x$

1. REJECT

$M_B$ on $x$

1. If $x = 1$, run $M$ on $w$
   - if $M$ accepts, ACCEPT
   - if $M$ rejects, loop

2. Else, REJECT

Since $M_A$ and $M_B$ are finite and each of their instructions is computable, $f$ is computable.

Suppose $\langle M, w \rangle \in A_{TM}$. Then $M$ accepts on input $w$. So $M_B$ accepts on input ‘1’ and rejects all other inputs. Since $M_A$ accepts no input, we have $|L(M_A)| = 0$ and $|L(M_B)| = 1$. Since $0 + 0 < 1$, $\langle M_A, M_A, M_B \rangle \in L$.

Suppose $\langle M, w \rangle \notin A_{TM}$. Then $M$ either rejects or loops on $w$. In either case, $M_B$ loops on ‘1’ and rejects all other input. So $|L(M_A)| = |L(M_B)| = 0$. Since $0 + 0 \not< 0$, $\langle M_A, M_A, M_B \rangle \notin L$.

• $\overline{A_{TM}} \leq_m L$

Define the reduction function $g$

$$g(\langle M, w \rangle) = \langle M_B, M_B, M_C \rangle$$

where $M_B$ is defined as above, and $M_C$ is defined as follows

$M_C$ on $x$

1. If $x = 1$, ACCEPT

2. Else, REJECT

Since $M_B$ and $M_C$ are finite and each of their instructions is computable, $g$ is computable.

Suppose $\langle M, w \rangle \in \overline{A_{TM}}$. Then $M$ either rejects or loops on $w$. Since $M_C$ accepts only on ‘1’, $|L(M_C)| = 1$ and, as above, $|L(M_B)| = 0$. Since $0 + 0 < 1$, $\langle M_B, M_B, M_C \rangle \notin L$.

Suppose $\langle M, w \rangle \notin \overline{A_{TM}}$. Then $\langle M, w \rangle \in A_{TM}$ and $M$ accepts on $w$. So, as above, $|L(M_C)| = |L(M_B)| = 1$. Since $1 + 1 \not< 1$, $\langle M_B, M_B, M_C \rangle \notin L$.

Therefore, $f$ is a reduction from $A_{TM}$ to $L$ and $g$ is a reduction from $\overline{A_{TM}}$ to $L$. So $L$ is co-unrecognizable.
III: \( L = \{ \langle M \rangle \mid M \text{ does not accept a string beginning with } 1 \} \)

Solution: (C) UNDECIDABLE and UNRECOGNIZABLE but its compliment is RECOGNIZABLE

We show \( L \) is undecidable by reducing it from \( \overline{A_{TM}} \). We then construct a recognizer for \( \overline{L} \).

- \( \overline{A_{TM}} \leq_m L \)
  
  Define the reduction function \( f \)
  
  \[
  f(\langle M, w \rangle) = \langle M' \rangle
  \]

  where \( M' \) is defined as follows
  
  \( M' \) on \( x \)
  
  1. If \( x = '1' \), run \( M \) on \( w \)
     
     - if \( M \) accepts, ACCEPT
     
     - if \( M \) rejects, loop
  
  2. Else, REJECT

  Since \( M' \) is finite and each of its instructions is computable, \( f \) is a computable function.

  Suppose \( \langle M, w \rangle \in \overline{A_{TM}} \). Then \( M \) either rejects or loops on \( w \). In either case, \( M' \) loops on \( '1' \) and rejects all other inputs. So \( M' \) does not accept a string beginning with \( '1' \), so \( \langle M' \rangle \notin L \).

  Suppose \( \langle M, w \rangle \notin \overline{A_{TM}} \). Then \( \langle M, w \rangle \in A_{TM} \) and \( M \) accepts on \( w \). So \( M' \) accepts on \( '1' \) and \( M' \) accepts some string beginning with \( '1' \). So \( \langle M' \rangle \notin L \).

- \( \overline{L} \) is recognizable.

  We construct a recognizer for \( \overline{L} \).

  \( R_{\overline{L}} \) on \( \langle M \rangle \)
  
  1. For \( i = 0 \) to \( \infty \)
     
     i. For all strings \( s \) of length \( \ell = 0 \) to \( i \) (ordered lexicographically)
        
        Run \( M \) on \( '1s' \) for \( i \) steps
        
        - if \( M \) accepts, ACCEPT

  Since \( R_{\overline{L}} \) is finite and each of its instructions is computable, \( R \) specifies a TM.

  Suppose \( \langle M \rangle \in \overline{L} \). Then \( M \) accepts some string \( '1n' \) in \( m \) steps. For all \( s \) and \( i \), \( R_{\overline{L}} \) runs \( M \) on \( '1s' \) for \( i \) (or more) steps. So the recognizer runs \( M \) on \( '1n' \) for \( m \) steps. So the recognizer accepts \( \langle M \rangle \).

  Suppose \( \langle M \rangle \notin \overline{L} \). Then \( \langle M \rangle \in L \) and \( M \) accepts no string beginning with \( '1' \). So, for no value of \( i \) and \( s \) does \( M \) accept on \( '1s' \) in \( i \) steps. So, \( R_{\overline{L}} \) loops and does not accept on \( \langle M \rangle \).

  Therefore, \( f \) is a reduction from \( A_{TM} \) to \( L \), so \( L \) is undecidable. Also, \( \overline{L} \) is recognizable.

  Since \( L \) is undecidable and \( \overline{L} \) is recognizable, \( L \) is unrecognizable.
IV: \( L = \{ \langle M, w, t \rangle \mid M \text{ accepts } w \text{ within } t \text{ steps} \} \)

Solution: (A) DECIDABLE

We construct a decider for \( L \).

\( D_L \) on \( \langle M, w, t \rangle \)

1. Run \( M \) on \( w \) for \( t \) steps
   
   if \( M \) accepts, ACCEPT
   
   if \( M \) does not accept, REJECT

Since \( D_L \) is finite and each of its instructions is computable, \( D_L \) specifies a Turing machine.

Suppose \( \langle M, w, t \rangle \in L \). Then \( M \) accepts on \( w \) in \( t \) steps. So \( D_L \) accepts on \( \langle M, w, t \rangle \).

Suppose \( \langle M, w, t \rangle \notin L \). Then \( M \) does not accept on \( w \) in \( t \) steps. So \( D_L \) rejects on \( \langle M, w, t \rangle \).

Therefore, \( D_L \) is a decider for \( L \) and \( L \) is decidable.
V: \( L = \{ \langle M \rangle \mid M \text{ accepts all strings beginning with 1} \} \)

Solution: (D) UNDECIDABLE and both it and its compliment are UNRECOGNIZABLE

We reduce \( L \) from \( A_{TM} \) and \( \overline{A_{TM}} \).

- \( A_{TM} \leq_m L \)
  
  Define the reduction function \( f \)

  \[ f(\langle M, w \rangle) = \langle M' \rangle \]

  where \( M' \) is defined as follows

  \( M' \) on \( x \)
  1. Run \( M \) on \( w \)
     - if \( M \) accepts, ACCEPT
     - if \( M \) rejects, loop

  Since \( M' \) is finite and each of its instructions is computable, \( f \) is a computable function.

  Suppose \( \langle M, w \rangle \in A_{TM} \). Then \( M \) accepts on \( w \) and \( M' \) accepts on all strings, including all strings that begin with ‘1’. So \( \langle M' \rangle \in L \).

  Suppose \( \langle M, w \rangle \notin A_{TM} \). Then \( M \) either rejects or loops on \( w \). In either case, \( M' \) loops on all inputs, and so loops on input ‘1’. So there is some input that starts with ‘1’ that \( M' \) does not accept, and \( \langle M' \rangle \notin L \).

- \( \overline{A_{TM}} \leq_m L \)
  
  Define the reduction function \( g \)

  \[ g(\langle M, w \rangle) = \langle M'' \rangle \]

  where \( M'' \) is defined

  \( M'' \) on \( x \)
  1. Run \( M \) on \( w \) for \( x \) steps
     - if \( M \) does not accept, ACCEPT
     - if \( M \) accepts, REJECT

  Since \( M'' \) is finite and each of its instructions is computable, \( f \) is a computable function.

  Suppose \( \langle M, w \rangle \in \overline{A_{TM}} \). Then \( M \) either rejects or loops on \( w \). So, for any value of \( x \), \( M \) does not accept \( w \) in \( x \) steps. So \( M'' \) accepts on all inputs, including all inputs that start with ‘1’. So \( \langle M'' \rangle \notin L \).

  Suppose \( \langle M, w \rangle \notin \overline{A_{TM}} \). Then \( \langle M, w \rangle \in A_{TM} \) and \( M \) accepts on \( w \). So, for some value of \( x \), \( M \) accepts \( w \) in at most \( x \) steps. So, there is some input \( s \) that \( M'' \) rejects. So \( M'' \) rejects ‘1s’ and there is some input that starts with ‘1’ that \( M'' \) does not accept. So \( \langle M'' \rangle \notin L \).

Therefore, \( f \) is a reduction from \( A_{TM} \) to \( L \) and \( g \) is a reduction from \( \overline{A_{TM}} \) to \( L \). So \( L \) is co-unrecognizable.
VI: $L(M)$ for some Turing machine $M$

Solution: (B) UNDECIDABLE but RECOGNIZABLE

If $L(M) = \phi$, then $L(M)$ is decidable. However, if $L(M) = A_{TM}$, then $L(M)$ is not. Without knowing more about the language, we cannot infer that it is decidable.

Claim: $M$ is a recognizer for $L(M)$.

Suppose $x \in L(M)$. Then, by definition, $M$ accepts on $x$.

Suppose $x \notin L(M)$. Then, by definition, $M$ does not accept on $x$.

Therefore, $M$ is a recognizer for $L(M)$ and $L(M)$ is recognizable.
VII: \( L = \{ \langle M, w, q \rangle \mid M \text{ enters a state } q \text{ during its computation of } w \} \)

Solution: (B) UNDECIDABLE but RECOGNIZABLE

We reduce \( L \) from \( A_{TM} \) and construct a recognizer for \( L \).

- \( A_{TM} \leq_m L \)
  
  Define the reduction function \( f \)
  
  \[
  f(\langle M, w \rangle) = \langle M', 1, q_A \rangle
  \]

  where \( M' \) is defined as follows

  \( M' \) on \( x \)
  
  1. Run \( M \) on \( w \)
     
     i. If \( M \) accepts, ACCEPT
     
     ii. Else if \( M \) rejects, loop

  and \( q_A \) is an accept state for \( M' \).

  Since \( M' \) is finite and each of its instructions is computable, \( f \) is a computable function.

  Suppose \( \langle M, w \rangle \in A_{TM} \). Then \( M \) accepts on \( w \). So \( M' \) accepts on all inputs, including ‘1’. Since \( M' \) accepts ‘1’, \( M' \) enters state \( q_A \) during its computation of ‘1’ (by definition of \( q_A \)). So \( \langle M', 1, q_A \rangle \in L \).

  Suppose \( \langle M, w \rangle \notin A_{TM} \). Then \( M \) either rejects or loops on \( w \). In either case, \( M' \) loops on all inputs, including ‘1’, and never enters its accept state \( q_A \). So \( \langle M', 1, q_A \rangle \notin L \).

- We now construct a recognizer for \( L \).

  \( R_L \) on \( \langle M, w, q \rangle \)
  
  1. Run \( M \) on \( w \)
     
     i. If \( M \) enters state \( q \), ACCEPT
     
     ii. Else if \( M \) accepts, REJECT
     
     iii. Else if \( M \) rejects, REJECT

  Since \( R_L \) is finite and each of its instructions is computable, \( R_L \) specifies a Turing machine.

  Suppose \( \langle M, w, q \rangle \in L \). Then \( M \) enters state \( q \) in its computation of \( w \). So \( R_L \) accepts.

  Suppose \( \langle M, w, q \rangle \notin L \). Then \( M \) does not enter state \( q \) in its computation of \( w \). So either \( M \) accepts before entering state \( q \), \( M \) rejects before entering state \( q \), or \( M \) loops and never enters state \( q \). In the first or second case, \( R_L \) rejects \( \langle M \rangle \). In the third case, \( R_L \) loops on \( \langle M \rangle \). So, in any case, \( R_L \) does not accept on \( \langle M \rangle \).

Therefore, \( f \) is a reduction from \( A_{TM} \) to \( L \) and \( R_L \) is a recognizer for \( L \). So \( L \) is recognizable but undecidable.