Theory of Computation	Units
Emanuele ”Manu” Viola	

1 Unit 1: Watch videos 1,2,3,4

Solve the following exercises

1. For each of the following assertions, indicate if it is always true (T), always false (F), or if it depends – that is, it can be either true or false depending on the meaning of A, B, C. (D).

   (a) $A \Rightarrow (B \Rightarrow A)$

   (b) $(\exists x A(x)) \Leftrightarrow \exists x(A(x) \wedge \forall y \neq x, \neg A(y))$

   (c) $(A \Rightarrow B) \Rightarrow (\neg B \Rightarrow \neg A)$

   (d) $A \Leftarrow ((B \Rightarrow A) \wedge B)$

   (e) $A \Leftarrow (B \Rightarrow A)$

   (f) $(A \vee B) \Rightarrow A$

   (g) $A \Rightarrow A$

   (h) "The earth is flat" $\Rightarrow$ "pigs can fly"

   (i) "$2^{16} + 1$ is prime" $\Rightarrow$ "$1 + 1 = 2$"

   (j) $(\neg A \vee A) \Leftrightarrow A$

   (k) $(A \wedge B) \Rightarrow A$

   (l) $A \Rightarrow (A \vee B)$

   (m) $(\neg A \Rightarrow A) \Leftrightarrow A$

   (n) $(\neg A \Leftarrow A) \Leftrightarrow A$

   (o) $(A \wedge (A \Rightarrow B) \wedge (B \Rightarrow C)) \Rightarrow C$

   (p) $((A \Rightarrow B) \wedge \neg B) \Rightarrow \neg A$

   (q) $(\neg(A \Rightarrow B)) \wedge B$

   (r) $A \Rightarrow (A \wedge B)$

   (s) $(A \vee B) \Rightarrow (A \Rightarrow \neg B)$

   (t) $(\forall i \in \{1, 2, 3, 4\} \exists k \in \{1, 2, 3\} : A(i, k))$
   $\wedge (\forall i, j \in \{1, 2, 3, 4\} \forall k \in \{1, 2, 3\} : \neg(A(i, k) \wedge A(j, k) \wedge i \neq j))$

2. Give the state diagrams of DFAs recognizing each of the following languages over the alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$.

   (a) $L_1 = \{w \mid w$ contains an odd number of $\mathtt{b}$'s$\}$

   (b) $L_2 = \{w \mid$ every $\mathtt{a}$ in w is immediately followed by a $\mathtt{b}\}$

   (c) $L_3 = \{w \mid$ every character in an even position in w is equal to w's first character$\}$. Define the first character to be at position 0, an even number.

(d) $L_4 = \{w \mid w \text{ is any string except } \texttt{aba}\}$

(e) $L_5 = \{w \mid w \text{ ends with } \texttt{bb}\}$

(f) $L_6 = \{w \mid w \text{ contains the substring } \texttt{aaa}\}$

3. Show that if A and B are regular languages, then so is the language $A \ominus B$. Give a detailed proof in the style of the videos. That is, given DFAs for A and B, define a DFA for $A \ominus B$, and prove its correctness according to the definition of accept.

4. Prove or disprove each of the following assertions. Throughout, the alphabet is $\Sigma = \{0, 1\}$:

   (a) There exists an infinite number of regular languages

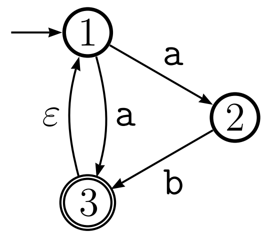
   (b) Every finite language is regular.

   (c) Every regular language is finite

2 Unit 2: Watch videos 5,6

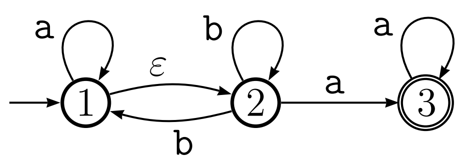
Solve the following exercises

1. Give the formal definition $(Q, \Sigma, \delta, q_0, F)$ of the following NFA, and show that it accepts the string `aab` according to the formal definition of accepting.

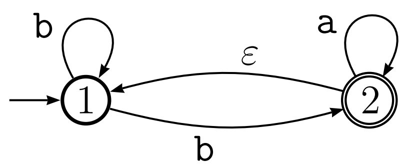


2. Convert each of the following NFAs to an equivalent DFA using the conversion process seen in class. Label each state of the NFA with the corresponding subset of states of the DFA.

(a)



(b)



3 Unit 3: Watch videos 7,8

Solve the following exercises

1. Give regular expressions that describe each of the following languages over the alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$.

   (a) $L_1 = \{w \mid \text{every } \mathtt{b} \text{ in } w \text{ is immediately followed by an } \mathtt{a}\}$

   (b) $L_2 = \{w \mid w \text{ contains the substring } \mathtt{bba}\}$

   (c) $L_3 = \{w \mid \text{every even position of } w \text{ is an } \mathtt{a}\}$

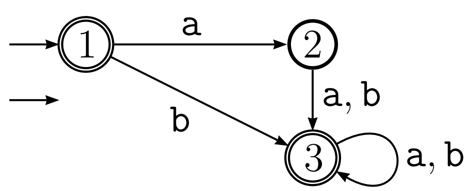
   (d) $L_4 = \{w \mid w \text{ begins with a } \mathtt{b} \text{ or ends with an } \mathtt{a}\}$

2. Convert each of the following regular expressions to an equivalent NFA using the conversion process seen in class. You need to show each step of the process.

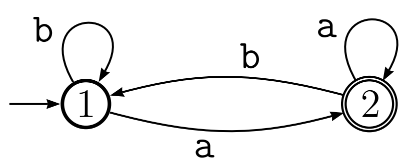
$$R_1 = \mathtt{a}^*(\mathtt{baa}^*)^* \qquad\qquad R_2 = (\mathtt{a} \cup \mathtt{b})^*\mathtt{b}$$

3. Convert each of the following DFAs to an equivalent regular expression using the conversion process seen in class. You need to show each step of the process.

   (a)



   (b)



4 Unit 4: Watch videos 9

Solve the following exercises

1. Use the pumping lemma to show that the following languages are not regular.

   (a) $L_1 = \{0^n 1^n \mid n \geq 0\}$

   (b) $L_5 = \{0^i 1^j 0^k \mid i, j, k \geq 0 \text{ and } i + j = k\}$

   (c) $L_6 = \{0^i 1^j \mid i, j \geq 0 \text{ and } i > j\}$

   (d) $L_7 = \{0^i 1^j \mid i, j \geq 0 \text{ and } 5i < j\}$

2. Let $\Sigma = \{a, b\}$ and L be the language of strings whose number of b is divisible by 5. Show that L cannot be recognized by a DFA with 4 states. Hint: The parameter p in the pumping lemma is the number of states of the DFA.

3. Give an alternative proof of the result that the language $L := \{0^n 1^n : n \geq 0\}$ is not regular without using the pumping lemma. Your proof should use the following principle (which you do not need to prove): If $f : A \to B$ is a function and $|A| > |B|$ then there are $a \in A$, $a' \in A$, $a \neq a'$ such that $f(a) = f(a')$. Hint: Let B be the set of states of the DFA.

5 Unit 5: Watch videos 10, 11

Solve the following exercises

1. Give context-free grammars for the following languages. Describe the language derived by each variable in your grammar.

   (a) $L_1 = \{0^n\, 1^n \mid n \geq 0\}$

   (b) $L_2 = \{0^i\, 1^j \mid i, j \geq 0 \text{ and } i > j\}$

   (c) $L_4 = \{0^m\, 1^m\, 0^n\, 1^n \mid m \text{ is even and } n \text{ is odd}\}$

   (d) $L_{10} = \{0^n 1^{2n} \mid n \geq 0\}$

   (e) $L_{11} = \{a^m b^n c^p \mid m + n = p\}$.

   (f) $L_{12} = \{a^m b^n c^p d^q \mid m + n = p + q\}$

2. Show that the following context-free grammars are ambiguous.

   (a) $S \rightarrow S + S \mid S \times S \mid 0 \mid 1$

   (b)
   $$\begin{aligned} S &\rightarrow T\,0\,T \mid V\,1\,V \\ T &\rightarrow 0 \mid 1 \mid \epsilon \\ V &\rightarrow 0 \mid 1 \mid \epsilon \end{aligned}$$

   (c)
   $$\begin{aligned} S &\rightarrow AB \\ A &\rightarrow 0\,A\,0 \mid 1\,A\,1 \mid 0 \mid 1 \\ B &\rightarrow BB \mid 0 \mid 1 \end{aligned}$$

6 Unit 6: Watch videos 12, 13, 14

Solve the following exercises

1. Use the pumping lemma to show that the following languages are not context-free.
   (a) $L_1 = \{\mathtt{a}^n\,\mathtt{b}^n\,\mathtt{c}^n \mid n \geq 0\}$
   (b) $L_2 = \{0^n\,1^n\,0^n\,1^n \mid n \geq 0\}$
   (c) $L_3 = \{0^i\,1^j\,2^k \mid i \geq j \geq k \geq 0\}$
2. Give a CFG for the language $L = \{w \in \{\mathtt{a}, \mathtt{b}\}^* \mid w \text{ has twice as many } \mathtt{a}\text{'s as } \mathtt{b}\text{'s}\}$. Show that your grammar is correct. For this problem you cannot use PDA.
3. Show that context-free languages are not closed under intersection. Hint: think of a language which we showed in class not to be context-free, and write it as the intersection of two context-free languages. Exhibit grammars for the context-free languages.

7 Unit 7: Watch videos 15

Solve the following exercises

1. Give the formal definition of a Turing machine M deciding a language L. Is your definition equivalent to the following definition? Why or why not?

   "M decides L if the following is true: for all w, M accepts $w \Leftrightarrow w \in L$."

2. Give the high-level description of a Turing machine that decides each of the following languages.

   (a) $L_3 = \{\mathtt{a}^i\,\mathtt{b}^j\,\mathtt{c}^k \mid i, j, k \geq 0 \text{ and } i + j = k\}$

   (b) $L_4 = \{ww^{\mathrm{R}} \mid w \in \{0, 1\}^*\}$
   (Recall that w^{R} denotes the reverse of w; for example, $0001011^{\mathrm{R}} = 1101000$.)

8 Unit 8: Watch videos 16, 17

Solve the following exercises

1. Show that each of the following languages is undecidable by reducing ATM to it.

   (a) $L_1 = \{M \mid M \text{ is a Turing machine and } L(M) = \emptyset\}$

   (b) $L_2 = \{M \mid M \text{ is a Turing machine and } L(M) \neq \Sigma^*\}$

   (c) $L_5 = \{(M, M') \mid M \text{ and } M' \text{ are Turing machines and } L(M) \subseteq L(M')\}$

2. In this problem you will show that you cannot decide anything about the language of Turing machines. Formally, let P be a set of languages over the alphabet $\Sigma = \{0, 1\}$. Let $L_P := \{M : M \text{ is a TM and } L(M) \in P\}$.

   Prove that

   (1) If $L_P = \emptyset$ then L_P is decidable.

   (2) If $L_P = \Sigma^*$ then L_P is decidable.

   (3) In all other cases, L_P is undecidable. For this proof you must reduce ATM to L_P.

3. In this problem you will fill the main missing detail of the proof that there is no decider for the language of CFG which derive Σ^*. Let M be a Turing machine with states Q and tape alphabet Γ. Let $\Sigma = Q \bigcup \Gamma$, and $\# \notin \Sigma$. Let $(a_i, b_i, c_i; d_i, e_i, f_i) : i = 1, 2, \ldots, t$ be the t 2x3 windows which are not consistent with M's transitions, as in the result about locality of computation.

   Give a CFG for the language $\{C\#D^R : C, D \in \Sigma^* \text{ and } C \text{ does not yield } D\}$, where D^R is the reverse of D. For each of the variables in your grammar, give the language it derives.

9 Unit 9: Watch videos 18, 19, 20, 21, 22

Solve the following exercises

1. For each of the following languages, first give the high-level description of a Turing machine that decides the language, then show an upper bound on the running time of your machine. Your upper bound should be tight up to constant multiplicative factors. In particular, if your machine runs in time n^2 and you show an upper bound of n^3, that is not sufficient.

   (a) $L_3 = \{\mathtt{a}^i\,\mathtt{b}^j\,\mathtt{c}^k \mid i, j, k \geq 0 \text{ and } i + j = k\}$

   (b) $L_4 = \{ww^{\mathrm{R}} \mid w \in \{0, 1\}^*\}$
   (Recall that w^{R} denotes the reverse of w; for example, $0001011^{\mathrm{R}} = 1101000$.)

   (c) $L_5 = \{\mathtt{a}^i\,\mathtt{b}^j\,\mathtt{c}^k \mid i, j, k \geq 0 \text{ and } i \cdot j = k\}$

   (d) $L_6 = \{ww \mid w \in \{0, 1\}^*\}$
   (Hint: Find the middle of the input string. Note that finding the middle is not a trivial operation, and you have to explain how it is accomplished in terms of head movements.)

2. Give a power-time reduction, and prove its correctness, from 3Sat to System, defined as follows. A *linear inequality* is an inequality involving sums of variables and constants, such as $x + y \geq z$, $x \leq -17$, and so on. A system of linear inequalities has an integer solution if it is possible to substitute integer values for the variables so that every inequality in the system becomes true. The language SYSTEM consists of systems of linear inequalities that have an integer solution. For example,

$$(x + y \geq z, x \leq 5, y \leq 1, z \geq 5) \in \text{SYSTEM}$$
$$(x + y \geq 2z, x \leq 5, y \leq 1, z \geq 5) \notin \text{SYSTEM}$$

3. The solitaire game STONES is played on an $n \times k$ grid of squares (n rows and k columns). In the initial configuration each square is occupied either by a green stone, or by a red stone, or is empty. You play the game by removing stones. You win if you can remove a set of stones so that:

   (1) Every row only has stones of the same color, and

   (2) Every column has at least one stone.

   Let STONES $= \{g : g$ is an $n \times k$ grid configuration that you can win$\}$. Reduce 3SAT to STONES in polynomial time; prove that your reduction works.

   (Hint: Given a 3SAT instance ϕ with n variables and k clauses, construct a configuration where the (i, j) square depends on variable x_i and clause C_j.)

4. For an integer k, k-COLOR is the problem of deciding if the nodes of a given undirected graph G can be colored using k colors in such a way that no two adjacent vertices have the same color.

   Give a polynomial-time reduction from 3-color to 4-color. Exhibit a reduction and prove that your reduction works.

10 Unit 10: Watch videos 23, 24, 25, 26

Solve the following exercises

1. Recall the language SYSTEM. Consider the following proof that SYSTEM is in NP: "Simply guess an assignment, and check that the inequalities are satisfied." Is this a correct proof?

2. For each of the following classes of languages, say if it remains the same if we were using JAVA as our computational model instead of Turing machines.

   (a) $\text{Time}\,(100n)$

   (b) P

   (c) NP

   (d) EXP

   (e) $\{L \mid L \text{ is decidable}\}$

3. This exercise will show that there is nothing 'special' or 'deep' about the existence of NP-complete problem. What is 'special' about 3SAT is that it is convenient for reductions, while the next problem is not.

   Without using the Cook-Levin theorem, prove that the following language is NP-complete:
   $L = \{(M, x, 1^t) : \ M \text{ is a Turing machine} : \exists y \in \{0,1\}^t : M(x,y) \text{ accepts within } t \text{ time steps}\}.$

4. Recall Regular Expressions with Exponentiation (REE). In what way is exponentiation useful in proving that the language All-REE $= \{R \mid R \text{ is an REE and } L(R) = \Sigma^*\}$ is not in P?

5. For each of the following results, say if the corresponding proof seen in class exploited the locality of Turing machines.

   (a) ATM is undecidable.

   (b) $\{(M, w) \mid M \text{ is a Turing machine that halts on input } w\}$ is undecidable.

   (c) All-CF $= \{G \mid G \text{ is a context-free grammar and } L(G) = \Sigma^*\}$ is undecidable.

   (d) All-REE $= \{R \mid R \text{ is an REE and } L(R) = \Sigma^*\}$ is not in P.

   (e) The Cook-Levin theorem.