

A. [Although no justification is required, I give justification, for educational purposes.]

1. TRUE. [This was a homework problem — 3.15, I think.]

2. TRUE. [If  $\bar{A}$  is finite, then  $\bar{A}$  is decidable, so  $A$  is decidable.]

3. TRUE. [A decider could easily inspect  $G$  to see how many edges it has.]

4. FALSE. [ $M$ 's immediate move is uniquely determined by the state and tape symbol read. However, the other contents of the tape or the location of the tape head may be different on the two occasions in question, so  $M$  may proceed differently after its immediate move.]

5. FALSE. [As we've learned in homework, two-stack PDAs are equivalent in power to Turing machines. So the statement is: If  $N$  is any Turing machine, then  $L(N)$  is decidable. But a language is recognizable if and only if it is the language of a Turing machine. So the statement is: If a language  $A$  is recognizable, then  $A$  is decidable. That's certainly false.]

6. FALSE. [This is a question about a property of recognizable languages:  $P(L)$  is " $L = L^{\text{rev}}$ ". This is a nontrivial property, so it is undecidable by Rice's theorem.]

7. TRUE. [In fact, the language is decidable. The decider converts the regular expressions to DFAs, and then runs the decider for  $EQ_{DFA}$  discussed in the textbook.]

8. FALSE. [See the definition of acceptance and rejection in non-deterministic Turing machines.]

B1. The transition is  $\delta : Q \times \Gamma \times \Gamma \times \Gamma \rightarrow Q \times \Gamma \times \Gamma \times \Gamma \times \{L, R\} \times \{L, R\} \times \{L, R\}$ . [The inputs to the function are a state and three tape symbols (as read from the three tapes). The outputs of the function are a state, three tape symbols (to be written to the three tapes), and three directions (for the three tape heads to move).]

B2. [I'll omit the drawing, but here are the ideas. You can get away with one state indicating "carry zero", one state indicating "carry one", and maybe one more state. The two numerals might not be of the same length. Blanks at the end of a numeral are equivalent to zeros. As soon as both summand symbols are blank, the algorithm can halt.]

C. Suppose, for the sake of contradiction, that  $R$  is a recognizer for  $A$ . We will use  $R$  to build a recognizer  $P$  for  $\overline{ACC_{TM}}$ . This Turing machine  $P$ , when given input  $\langle M, w \rangle$ , proceeds as follows.

1. Build a Turing machine  $N$  that accepts all inputs.

2. Run  $R$  on  $\langle M, N, w \rangle$ , and output whatever  $R$  outputs.

If  $\langle M, w \rangle \in \overline{ACC_{TM}}$ , then  $M$  does not accept  $w$ , while  $N$  does accept  $w$ . Thus  $R$  accepts  $\langle M, N, w \rangle$ , and  $P$  accepts  $\langle M, w \rangle$ . On the other hand, if  $\langle M, w \rangle \in ACC_{TM}$ , then  $M$  and  $N$  both accept  $w$ , so  $R$  does not accept  $\langle M, N, w \rangle$ , so  $P$  does not accept  $\langle M, w \rangle$ . Thus  $P$  is a recognizer for  $\overline{ACC_{TM}}$ . However,  $\overline{ACC_{TM}}$  is not recognizable, as we've discussed in class. This contradiction implies that  $A$  is not recognizable either.

D. [This problem is taken almost verbatim from our “practice exam”. I’ll give two proofs.]

Suppose, for the sake of contradiction, that  $A$  is decidable. Let  $H$  be a decider for  $A$ . We will use  $H$  to build a Turing machine  $N$  that we know cannot exist. On input  $\langle D, M \rangle$ ,  $N$  proceeds as follows:

1. Check that  $D$  is a DFA and  $M$  is a Turing machine. If not, reject. If so, continue...
2. Process all strings  $x$  in lexicographic order. For each  $x$ :
  - (a) If  $x$  is not a valid DFA encoding  $\langle C \rangle$ , then proceed to the next  $x$ . Otherwise...
  - (b) Run  $H$  on  $\langle D, M, C \rangle$ . If  $H$  rejects, then proceed to the next  $x$ . Otherwise...
  - (c) Rewrite the tape to contain  $\langle C \rangle$  (and nothing else), and accept.

We know from our homework that, for any  $D$  and  $M$ , the language  $L(D)/L(M)$  is regular. Hence there exists some DFA  $C$  such that  $L(C) = L(D)/L(M)$ . Our Turing machine  $N$  will eventually find this  $C$  and output the encoding  $\langle C \rangle$  on its tape. But we also know from homework that no such  $N$  can exist (because it could be used to build a decider for  $EMPTY_{TM}$ , which is not decidable). This contradiction implies that  $A$  is not decidable.

Suppose, for the sake of contradiction, that  $A$  is decidable. Let  $H$  be a decider for  $A$ . We will use  $H$  to build a decider  $N$  for  $EMPTY_{TM}$ . This Turing machine  $N$ , on input  $\langle M \rangle$ , proceeds as follows:

1. Build a DFA  $D$  that accepts all inputs, and a DFA  $C$  that rejects all inputs.
2. Run  $H$  on  $\langle D, M, C \rangle$ , and output whatever  $H$  outputs.

If  $L(M) = \emptyset$ , then  $L(D)/L(M) = \emptyset = L(C)$ , so  $H$  accepts  $\langle D, M, C \rangle$  and  $N$  accepts  $\langle M \rangle$ . On the other hand, if  $L(M) \neq \emptyset$ , then  $L(D)/L(M) \neq \emptyset = L(C)$ , so  $H$  rejects  $\langle D, M, C \rangle$  and  $N$  rejects  $\langle M \rangle$ . Thus  $N$  decides  $EMPTY_{TM}$ , which is not decidable. This contradiction implies that  $A$  is not decidable either.