

# FINAL

CIS 130 - Spring 08

Manfred Warmuth

NAME: Solutions

Student ID: \_\_\_\_\_

This exam is closed book and closed notes.

Show partial solutions to get partial credit.

If your questions are not written legibly, you won't get full credit.

Clarity and succinctness will be rewarded!

Question 1: \_\_\_\_\_(out of 20)

Question 2: \_\_\_\_\_(out of 10)

Question 3: \_\_\_\_\_(out of 10)

Question 4: \_\_\_\_\_(out of 10)

Question 5: \_\_\_\_\_(out of 10)

Question 6: \_\_\_\_\_(out of 10)

Question 7: \_\_\_\_\_(out of 10)

Question 8: \_\_\_\_\_(out of 10)

Total: \_\_\_\_\_(out of 100)

Total EC: \_\_\_\_\_(out of 10)

1. Short questions:

(a) What is the Church-Turing Thesis?

2

Computations on any computational device can be simulated on a TM

(b) Is there an unrestricted grammar generating the following language?

$$SA = \{e(T) : T \text{ is TM that accepts } e(T)\}.$$

Give a two-sentence reason of whether this is true or not.

Yes since

- SA T.a.

-  $\forall$  T.a. lang.  $L \exists$  unrestricted grammar  $G$  s.t.

$$L(G) = L$$

(c) What is the language associated with the Halting problem?

$$H = \{e(T) e(w) : \text{TM } T \text{ halts on input } w\}$$

↑  
runs into  $h_a$  or  $h_r$

(d) Show that Turing acceptable languages are closed under finite union.

Assume  $T_i$  accepts  $L_i$  ( $1 \leq i \leq k$ )

Then the following TM's accept  $\bigcup_{i=1}^k L_i$

- Guess  $1 \leq i \leq k$  & run  $T_i$
- Simulate all  $T_i$  in parallel on the input
  - one step at a time
  - as soon as one  $T_i$  accepts then accept

- (e) Give a language that is not Turing acceptable.

$$NSA = \{ e(T) : TM T \text{ does not accept } e(T) \}$$

- (f) Are non-deterministic Turing machines more powerful than deterministic Turing machines? Why or why not?

No

Non-det. TM's can be simulated  
by det. TM's

- (g) Give a closure property that holds for Turing decidable languages that does not hold for Turing acceptable languages.

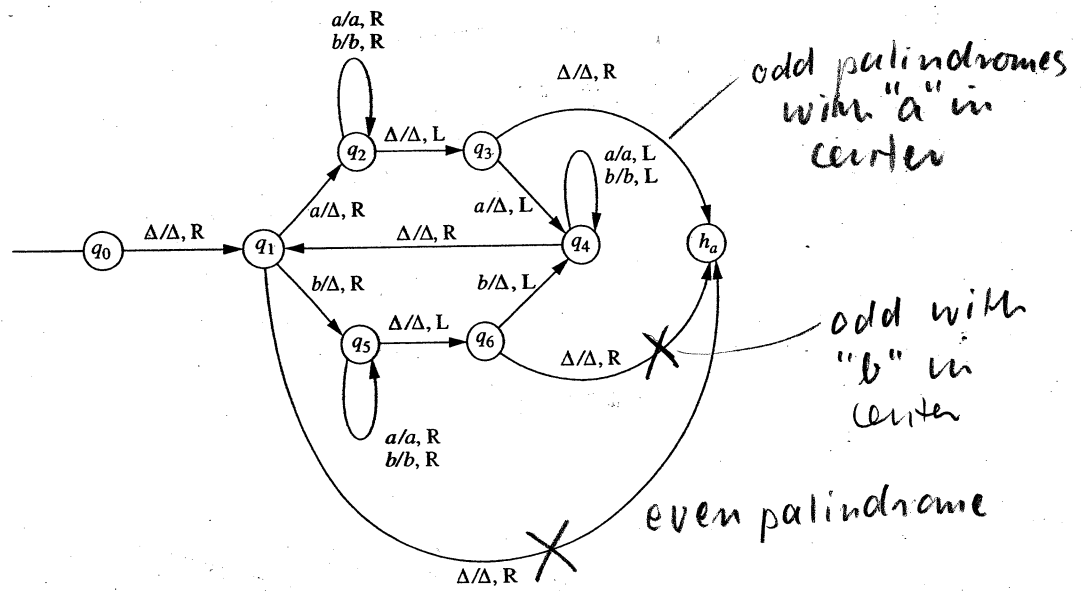
complementation or set difference

2. The following Turing machine accepts the language of palindromes over  $\{a, b\}$ :  $\{w : w = w^R\}$ . Modify the machine so that it accepts only palindromes of odd length that have an 'a' in the middle position.

Examples of such palindromes are:

abbabba aaa baaab

Hint: You need to delete two transitions.

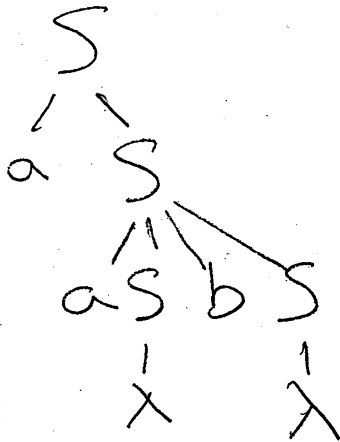


3. Show that the following grammar is ambiguous:

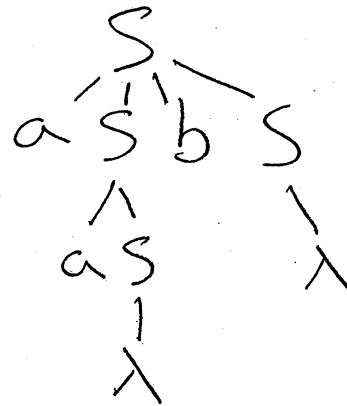
$$S \rightarrow aS \mid aSbS \mid \lambda$$

Hint: A three letter word suffices!

aab



$$S \rightarrow aS \rightarrow aaSbS \rightarrow aab$$



$$S \rightarrow aSbS \rightarrow aaSb \rightarrow aab$$

4. Give a short proof that context free languages are not closed under intersection.

Hint: Do a proof by contradiction.

$$L_1 = \{ a^i b^j c^k \mid i=j, i, j, k \geq 0 \}$$

$$L_2 = \{ a^i b^j c^k \mid j=k, i, j, k \geq 0 \}$$

$L_1$  &  $L_2$  are CFL's

$$L_1: S \rightarrow AC$$

$$A \rightarrow aAb \mid \lambda$$

$$C \rightarrow Cc \mid \lambda$$

$$L_2: S \rightarrow AB$$

$$A \rightarrow aA \mid \lambda$$

$$B \rightarrow bBc$$

Proof: Assume CFLs closed under intersection

This implies for CFL's  $L_1$  &  $L_2$

Then  $L_1 \cap L_2$  is a CFL.

choose  $L_1$  &  $L_2$  as above.

$$L_3 = L_1 \cap L_2 = \{ a^i b^j c^k \mid i=j=k \}$$

But we know from class  $L_3$  is not a CFL.

We can prove with pumping lemma using  $L_3 = a^n b^n c^n$

This is a contradiction!

$\Rightarrow$

Intersection not closed for CFLs

↳

#

5)

Convert the following context-free grammar into Chomsky Normal Form. Show your steps!  
The terminal letters are  $a$  and  $b$ .

$$\begin{aligned} S &\rightarrow ASB \mid \lambda \\ A &\rightarrow aAS \mid a \\ B &\rightarrow SbS \mid A \mid bb \end{aligned}$$

✓ 1) Eliminate  $\lambda$  productions

$$\begin{aligned} S &\rightarrow ASB \mid AB \\ A &\rightarrow aAS \mid aA \mid a \\ B &\rightarrow SbS \mid bS \mid sb \mid b \mid A \mid bb \end{aligned}$$

✓ 2) Eliminate unit productions

$$\begin{aligned} S &\rightarrow ASB \mid AB \\ A &\rightarrow aAS \mid aA \mid a \\ B &\rightarrow SbS \mid bS \mid sb \mid aAS \mid aA \mid a \mid b \mid bb \end{aligned}$$

✓ 3) Restrict to terminals and non terminals (single terminals / nonterminal groups)

$$\begin{aligned} S &\rightarrow ASB \mid AB \\ A &\rightarrow X_aAS \mid X_aA \mid a \\ B &\rightarrow SX_bS \mid X_bS \mid SX_b \mid X_aAS \mid X_aA \mid a \mid b \mid X_bX_b \\ X_a &\rightarrow a \quad X_b \rightarrow b \end{aligned}$$

✓ 4) Limit length

$$\begin{aligned} S &\rightarrow AT_1 \mid AB \\ T_1 &\rightarrow SB \\ A &\rightarrow X_aT_2 \mid X_aA \mid a \\ T_2 &\rightarrow AS \\ B &\rightarrow ST_3 \mid X_bS \mid SX_b \mid X_aT_2 \mid X_aA \mid X_bX_b \mid a \mid b \\ T_3 &\rightarrow X_bS \\ X_a &\rightarrow a \\ X_b &\rightarrow b \end{aligned}$$

7

6. Give a complete proof that the following language is not context free:

$$\{a^n b^n a^n : n \geq 0\}$$

Hint: Use the following version of the pumping lemma for context free languages. Think carefully about whether to pump up or down!

If  $L$  is context free, then there exists  $N$  such that: If  $u \in L$  and  $|u| \geq N$ , then  $u$  can be written as  $vwxyz$  for which

- $|wxy| \leq N$ ,
- $|wy| > 0$ , and
- $vw^mxy^mz \in L$  for all  $m \geq 0$ .

Assume  $L$  cf.

Then PL applies.

Let  $N$  be constant of PL.

Choose  $u = a^N b^N c^N$

By PL  $u$  can be written as  $vwxyz$  s.t.

a) - c) applies.

Since  $|wxy| \leq N$ ,  $wy$  misses first or last block of "a"s.

Since  $|wy| > 0$ ,  $vxy$  has less letters in one or two blocks and  $N$  letters in the first or last block.

This means  $vxy \notin L$   
contradicting the PL

We conclude that  $L$  not cf.

□

7. Can the following grammar

$$\begin{aligned}
 S &\rightarrow AB \mid AC \mid BC \\
 A &\rightarrow AA \mid AB \mid a \\
 B &\rightarrow BB \mid BC \mid b \\
 C &\rightarrow CC \mid AC \mid a
 \end{aligned}$$

generate the string *babaa*? Use the CYK algorithm. Show your work by giving the table. How can the table be used to answer the question whether the string can be generated or not!

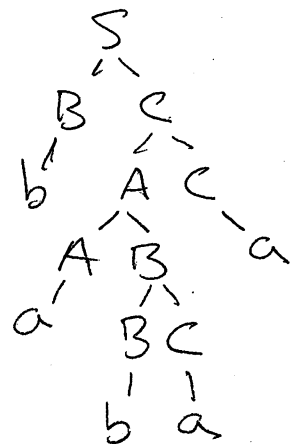
b a b a a

|     |       |     |      |     |
|-----|-------|-----|------|-----|
| B   | A,C   | B   | A,C  | A,C |
| S,B | S,A   | S,B | A,SC |     |
| B   | S,A,C | S,B |      |     |
| S,B | S,A,C |     |      |     |
| S,B |       |     |      |     |

Yes the String can be generated because we have an S in the final box.

one generation would be

$S \rightarrow BC \rightarrow bAC \rightarrow bABa \rightarrow baBCa \rightarrow babaa$



8. a) Does the following grammar generate a word in  $a^*$ ?

$$\begin{aligned} S &\rightarrow ABaTc \mid BB \\ B &\rightarrow BaA \mid b \mid aA \\ A &\rightarrow BA \mid \lambda \\ T &\rightarrow aaa \end{aligned}$$

If so then give a derivation tree for a word in  $a^*$ .

b) You are to sketch an algorithm for deciding the following problem:

Input: A context-free grammar  $G$ .

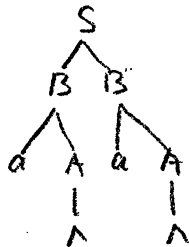
Question: Does  $G$  generate a word in  $a^*$ ?

That is,  $L(G) \cap a^*$  is non-empty?

Reason that your algorithm runs in polynomial time.

Hint: Which productions can never be used in the derivation for a word in the language  $a^*$ ?

a)  $aa \in L(G)$



1. Eliminate all productions with a terminal  $\neq "a"$
2. Replace all "a"s in remaining productions by  $\lambda$
3. If  $S$  nullable then answer yes

1) and 2), simple scans of the grammar that are poly in the size of grammar

3), poly time to check for nullability

g. Extra Credit: Prove that the Halting problem is undecidable.

3 proofs given in  
second last lecture.