Cpt S 317 Final Exam

WSU property. Please do not distribute.

1. (10 pts) Convert the following context free grammar into Chomsky Normal Form:

$S \to AAB|abc$
$A \to ab$
$B \to bc$

---

2. (10 pts) Write a context-free grammar generating language $\{a^{2n}b^{3m}a^{2m}b^n : n > 0, m > 0\}$.

---

3. (10 pts) Write a context-free grammar generating language $\{a^n b^m a^k : 3n + 4m < k, n \geq 0, m \geq 0, k \geq 0\}$.

---

3. (10 pts) Describe in English a pushdown automaton accepting $\{a^{2n}b^{3m}a^{2m}b^n : n > 0, m > 0\}$.

---

4. (10 pts) Let L be a context free language. Show that it is decidable whether there is a word in L with odd length. (Hint: the emptiness problem for context-free language has an algorithm to solve, as we have learned.)

---

5. (10 pts) Show that $\{a^n b^n c^{2n} : n > 0\}$ is not a context-free language.

---

6. (10 pts) Build a Turing machine (by writing the δ-instructions) to recognize the following language: $a(aa)^*b$.

---

7. (10 pts) Show that there are infinitely many Turing machines recognizing the following language: $a(aa)^*b$.

---

8. (10 pts) Show that if L_1 is r.e. and L_2 is recursive, then $L_1 \cup \bar{L}_2$ is r.e. ($\bar{L}_2$ is the complement of L_2)

---

9. (10 pts) Suppose that L is a regular language over alphabet $\{0, 1\}$. Define $\heartsuit$ to be the following language: $\{xx^r : \exists y \quad xy \in L, \ |x| = |y|\}$. Show that

$\heartsuit$ is context-free. (x^r is the reverse of word x; e.g., when x is abc, the x^r is cba)

---

10. (10 pts) Describe the language generated by the following grammar: $S \to 11S00|0$.

---

11. (10 pts) Please mark true/false to each statement; no explaination is needed.

a. A subset of a recursive language is also recursive.
b. A subset of a context-free language is also context-free.
c. An algorithm is a Turing machine that halts on all input.
d. An r.e. language can not be a subset of a regular language.
e. There is an r.e. language that is not recursive.
f. Halting problem of C-programs is decidable.
g. Halting problem of Turing machines is undecidable.
h. Halting problem of PDA's is decidable.
i. Alan Turing invented Turing machines.
j. PDAs can be nondeterministic.

---

12. (10 pts) Let L be a recursive language. Show that the following is also a recursive language:

$$\{x : \exists y.\ |x| = |y|, xy \in L\}.$$

---

13. (10 pts) Let L be an r.e. language. Show that the following is also an r.e. language:

$$\{x : \exists y.\ xy \in L\}.$$

6. Build a Turing machine to recognize the following language: $a(aa)^*b$

* State q0:
  on input a, go to state q1
  on input b, go to state q2
  on any other ~~state~~ input, reject the input
  State q1:
  on input a, go to state q1
  on input b, go to state q3
  on any other input, reject the input
  State q2:
  on any input, reject the input
  State q3:
  on any input, accept the input

* This Turing machine uses four states, in q_0 it reads the input one character at a time. If it sees an a it goes to state q1, if it sees b it goes to state q2 otherwise rejects the input

7. Show that there are infinitely many Turing machines recognizing the following language: $a(aa)^*b$

* Regular set for that regular expression is {aaab, aaaaab, aaaaaaab, aaaaaaaaab, ...}
  We can form as many regular values, but it should start with 'a' and end with 'b' in between then it can have any number of (aa)'s, Hence there are infinitely many Turing machines recognizing the following language $a(aa)^*b$

8. Show that if L_1 is r.e. and L_2 is recursive, then $L_1 \cup \bar{L}_2$ is r.e. ($\bar{L}_2$ is the complement of L_2)

logic:
- L_1 is recursively enumerable and L_2 is recursive using closure property of recursive languages, so $\bar{L}_2$ (complement of L2) is also recursive.
- ~~...~~ every recursive language is also recursively enumerable, so $\bar{L}_2$ is also recursively enumerable
- ~~so~~ Recursively enumerable languages are closed under union
- $L_1 \cup \bar{L}_2$ is also recursively enumerable

9. Suppose that L is a regular language over alphabet {0, 1}. Define R to be the following language: $\{xx^r : \exists_y\ xy \in L, |x| = |y|\}$
   Show that R is context-free

- If Q is the number of states in M, then M_1 will have $Q_1 = Q \times Q$ states
- the transition of function M_1 will be $((q_1, q_2, a, b) \rightarrow (q_3 q_4, c, d, x_1, x_3)$
- In the end TM M_1 will accept the input if and only if both the tape content is accepted by M. So in this way M_1 will be TM which will always give correct result, since TM exists for L_1 will always give correct result and halt, hence the corresponding language L1 is recursive

10. Describe the language generated by the following grammar
    S → 11S0010

* language L = {11000, 111100000, 11111100000000, 11111111 0000000000}

  This grammar generates strings of n even number of 1's followed by (n+1) number of 0's

  $L = \{w \mid 1^n 0^{n+1}$, where $n >= 2$ and n is even}

11.
a. True e. True i. True
b. True f. False j. True
c. False g. True
d. True h. False

12. Let L be a recursive language. Show that the following is also a recursive language: $\{x : \exists_y, |x| = |y|, xy \in L\}$

- Let x be a recursive language and y be a r.e
- per the first term that is that means some value of y can be x
- it shows that x can be reducible to y which means a many to one reduction is possible. both x and y belong to the language L
- After reduction, the |x| = |y| is possible
* Therefore, the specified expression is also recursive language

13. * To show that the language ~~..~~ L =
- Let ~~..~~ A1 and A2 be finite automata for L
- Let Q_1 and Q_2 be the state sets of A1 and A2 respectively
- The start set of the new automata for L' is $Q = Q_1 \times Q_2$
- The initial state of the new automaton $q_{1,0}, q_{2,0}$
- For each accepting state q in A1 and each accepting state q' in A2 add the state

* The automaton can recognize the L', so L' is regular

Mark Shinozaki

1. Convert the following context free grammar into Chomsky Normal Form:

$S \to AAB \mid abc$

$A \to ab$

$B \to bc$

* CNF is the form of grammar in which the production rules are in the form

$A \to BC$

$A \to a$

1 *

$S \to AABS \mid abc$
$A \to ab$
$B \to bc$

↓

$S' \to S$
$S \to AABS \mid abc$
$A \to ab$
$B \to bc$

↓

$S' \to AABS \mid abc$
$S \to AABS \mid abc$
$A \to ab$
$B \to bc$

2 *

$S' \to AU \mid abc$
$S \to AU \mid abc$
$A \to ab$
$B \to bc$
$U \to ABS$

↓

$S' \to AU \mid abc$
$S \to AU \mid abc$
$A \to ab$
$B \to bc$
$U \to AV$
$V \to BS$

$S' \to AU \mid aB$
$S \to AU \mid ab$
$A \to ab$
$B \to bc$
$U \to AV$
$V \to BS$

↓

$S' \to AU \mid aB$
$S \to AU \mid aB$
$A \to aX$
$B \to bY$
$U \to AV$
$V \to BS$
$X \to b$
$Y \to c$

$S' \to AU \mid ZB$
$S \to AU \mid ZB$
$A \to ZX$
$B \to XY$
$U \to AV$
$V \to BS$
$X \to b$
$Y \to c$
$Z \to a$

↓

CNF is of form
$A \to BC$ or $A \to a$

* $S' \to AU \mid ZB$ *
$S \to AU \mid ZB$
$A \to ZX$
$B \to XY$
$U \to AV$
$V \to BS$
$X \to b$
$Y \to c$
$Z \to a$

2. Write a context-free grammar generating language $\{a^{2n} b^{3m} a^{2m} b^n : n > 0, m > 0\}$

* Grammar:

$S \to aaSb \mid aaAb$

$A \to bbbAaa \mid bbbaa$

a simple string with $n = 1, m = 1$:

$a^{2n} b^{3m} a^{2m} b^n = aabbbaab$ $\quad S \to aaAb \to aabbbaab$

3. Write a context-free grammar generating language $\{a^n b^m a^k : 3n + 4m < k, n \geq 0, m \geq 0, k \geq 0\}$

* $S \to aX\, bY aZ$ $X \to aX \mid \varepsilon$ $Y \to bY \mid \varepsilon$ $Z \to aZ \mid \varepsilon$ *

3. Describe in English a pushdown automaton accepting $\{a^{2n} b^{3m} a^{2m} b^n : n > 0, m > 0\}$

~~[illegible]~~

* push A for two a's

* for next every two a's push one A

* push B for 3 b's

* For next every three b's push one B

* pop B for every two a's - repeat as long as a's are there

* pop A for every b - Repeat as long as b's are there

* Accept if the stack has base symbol

4. Let L be a context free language. Show that it is decidable whether there is a word in L with odd length.

For example if the original grammar contains a rule of the form $A \to BCD$, we can add a new rule of the form $A' \to BCD'$ to generate words with odd length. Once the grammar is constructed we can apply the algorithm for solving the emptiness problem to determine whether ~~[illegible]~~ the language L' is empty or not

5. Show that $\{a^n b^n c^{2n} : n > 0\}$ is not a context-free language.

1. let x be a string which belongs to the given language and length of x is greater than or equal to n, $|x| >= n$

Case 1: VWX has no A'S

& Case ~~[illegible]~~ VWX has no c's is similar gives us contradiction

2. now break S into 5 parts $x = uvwxy$ if any conditions fail this is not CFL

a. $|vwx| <= n$

b. $|vx| >= 1$

c. $u v^i w x^i y$ should belong to L for all $i >= 0$

- We have provided a case for CFL and contradicted was found.