Comp 330 (Fall 2024): Assignment 3

Answers must be submitted online by November 15 (11:59 pm), 2024.

General instructions

- **Important:** All of the work you submit must be done by only you, and your work must not be submitted by someone else. Plagiarism is academic fraud and is taken very seriously.
- To some extent, collaborations are allowed. These collaborations should not go as far as sharing code or giving away the answer. **You must indicate on your assignments the names of the people with whom you collaborated or discussed your assignments (including members of the course staff). If you did not collaborate with anyone, write "No collaborators". If asked, you should be able to orally explain your solution to a member of the course staff.**
- It is your responsibility to guarantee that your assignment is submitted on time. We do not cover technical issues or unexpected difficulties you may encounter. Last minute submissions are at your own risk.
- Multiple submissions are allowed before the deadline. We will only grade the last submitted file. Therefore, we encourage you to submit as early as possible a preliminary version of your solution to avoid any last minute issue.
- Late submissions can be submitted for 24 hours after the deadline, and will receive a flat penalty of 20%. We will not accept any submission more than 24 hours after the deadline. The submission site will be closed, and there will be no exceptions, except medical.
- In exceptional circumstances, we can grant a small extension of the deadline (e.g. 24h) for medical reasons only. However, such request must be submitted before the deadline, justified and approved by the instructors.
- Violation of any of the rules above may result in penalties or even absence of grading. If anything is unclear, it is up to you to clarify it by asking either directly the course staff during office hours, by email at or on the discussion board on Ed. Please, note that we reserve the right to make specific/targeted announcements affecting/extending these rules in class and/or on the website. It is your responsibility to monitor Ed for announcements.
- The course staff will answer questions about the assignment during office hours or in the online forum. We urge you to ask your questions as early as possible. We cannot guarantee that questions asked less than 24h before the submission deadline will be answered in time. In particular, we will not answer individual emails about the assignment that are sent the day of the deadline.
- Unless specified, **you must show your work and all answers must be justified.**

1. (20 points) Let G be a context-free grammar (CFG) in a Chomsky Normal Form (CNF).

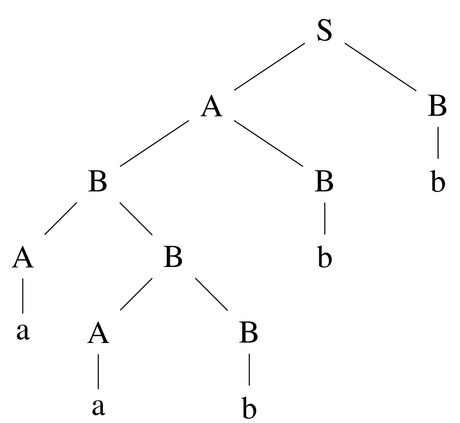
$$\begin{aligned} S &\rightarrow AB \\ A &\rightarrow BB \mid a \\ B &\rightarrow AB \mid b \end{aligned}$$

Use the Cocke-Younger-Kasami (CYK) algorithm to decide if the string $s = aabbb$ is a word of the language $L(G)$. If yes, draw a derivation tree for s computed from your execution of the CYK algorithm. Highlight the symbols selected in the CYK tables.

Solution:

$$\begin{array}{ccccc}
\{\mathbf{S}, B\} & & & & \\
\{\mathbf{A}\} & \{S, B\} & & & \\
\{S, \mathbf{B}\} & \{A\} & \{S, B\} & & \\
\emptyset & \{S, \mathbf{B}\} & A & A & \\
\{\mathbf{A}\} & \{\mathbf{A}\} & \{\mathbf{B}\} & \{\mathbf{B}\} & \{\mathbf{B}\} \\
\hline
a & a & a & b & b
\end{array}$$

The start symbol S is found in the top cell and therefore $s \in L(G)$. The derivation tree is:

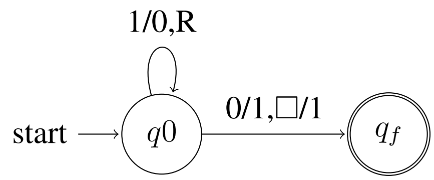


2. (10 points) Using the pumping lemma for regular languages, show that the language L of words over the alphabet $\Sigma = \{a\}$ whose length is a prime number is not a regular language.

Solution: Assume L is a rational language and let k be the number of the pumping lemma. Soit $x \in L$ such that $|x| \geq k$. Therefore, $\exists u, v, w$ such that $x = uvw$ and $|u| + |v| + |w|$ is a prime number. And $\forall i \geq 0$, $uv^iw \in L$. In particular, it is true for $i = |u| + |w|$, thus $y = uv^{|u|+|w|}w \in L$, and its length is a prime number. But $|y| = |u| + (|v| \cdot (|u| + |w|)) + |w| = (|v| + 1) \cdot (|u| + |w|)$ is not a prime number. Contradiction. Thus L is not regular.

3. (10 points) Describe a Turing machine that computer and write $n + 1$ for a input number n written binary notation on the input tape. Here, we assume that the binary number is written from left to right from the least significant bits to the most significant bits.

Solution: To add 1, we simply need to either change the least significant bit to one if it was zero (or if there was none), or to change it to zero if it was one and in this case, since there is a carry of one, add it to the binary number formed by the other bits. The solution is shown in the following automaton. We do not put the head back to the beginning.



4. (30 points) Consider the following language

$$L = \{w\#w : w \in \{a,b\}^*\}$$

Show that this language is decidable by

1. Providing a high-level description of a deterministic infinite tape Turing machine M which decides it.
2. Providing the state transition diagram of M.

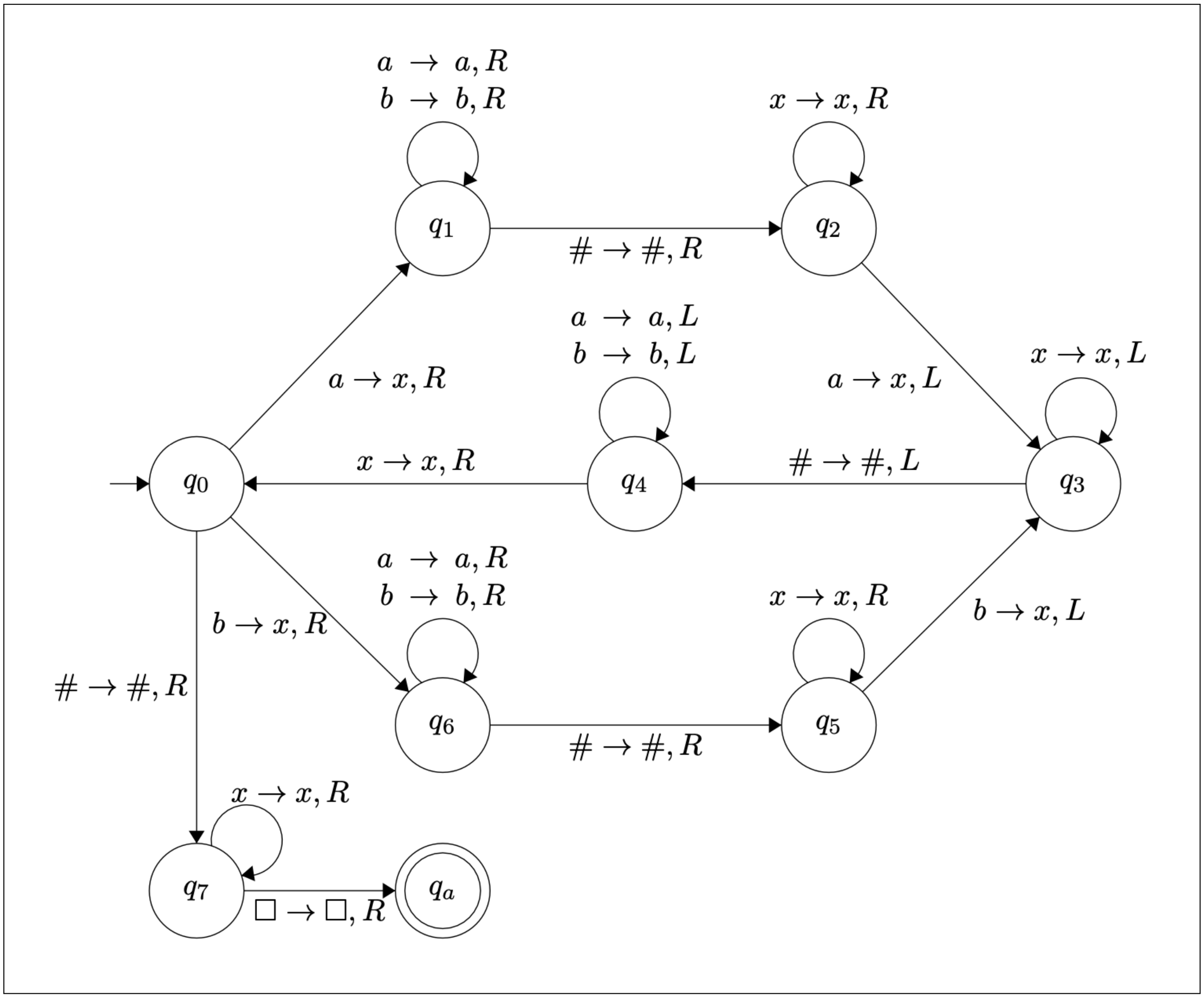
You do not need to prove the correctness of your construction.

Solution: M := On input w

1. If the pointer sees a #, it moves right through all the x's. If a letter (a or b) is encountered, M rejects. Otherwise, when a $\square$ is reached, M accepts.

2. Otherwise, M marks the letter (either a or b) with an x and moves the pointer right until it reaches #.

3. Past the #, M moves the pointer through all the x's until it sees the first un- marked letter. If the unmarked letter matches with the one from 2., M moves the pointer left, through the # until it reaches the first marked x on the left. M moves the pointer right and repeats from step 1. Otherwise, rejects.



5. (30 points) Let $\Sigma = \{0\}$ and consider the following function $f : \Sigma^+ \to \Sigma^+ f(x) = x \cdot x \cdot x$. Show that f is a total computable function by

   1. Providing a high-level description of a deterministic infinite tape Turing machine M which computes f.
   2. Providing the state transition diagram of M.

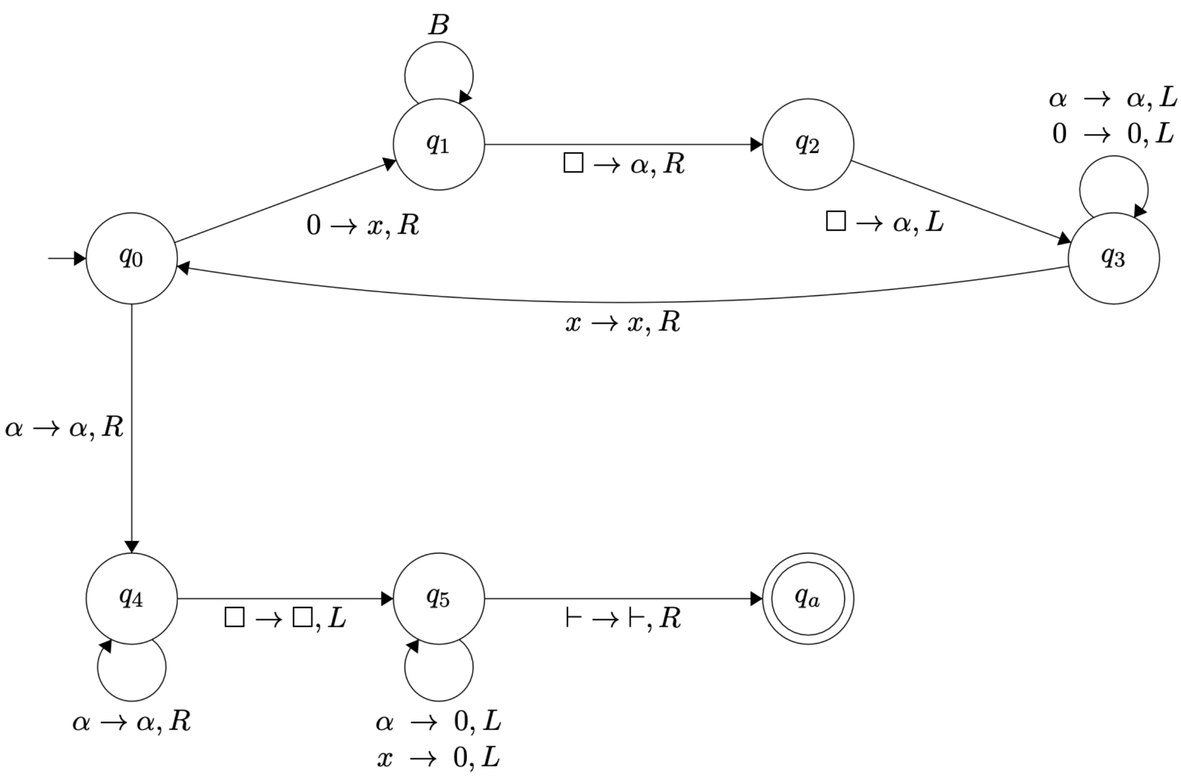
   You do not need to prove the correctness of your construction.

Solution: M := On input w

1. If the pointer reads a 0, M marks it with an x and moves right until it reaches a $\square$. M then continues to step 2. Otherwise, if the pointer reads an α, M moves the pointer all the way to the right until it reaches the first $\square$ and then moves the pointer left converting α and x to 0. Once it reaches $\vdash$, M moves its pointer one step to the right and returns.

2. Once a $\square$ is reached, M writes α twice and then moves the pointer left until it reaches the first marked x.

3. M moves the pointer one step to the right and repeats from step 1.



Question:	1	2	3	4	5	Total
Points:	20	10	10	30	30	100
Score:						