COSC 4200
Computability and Complexity

Name: Jacob Tuttle

Homework 04

March 12, 2020

1. Suppose ADD is regular. Let p be the pumping constant for ADD. Let $w = a^p b^p c^{2p}$. Then w can be broken up into substrings xyz where $|xy| \leq p$ and $0 \leq |y|$ such that $xy^i z$ exists for all i.

   Then we know that $x = a^k, y = a^j$, and $z = a^{p-k-j} b^p c^{2p}$ where $k \geq 0, j > 0$.

   Let $i = 2$. Then:

   $$\begin{aligned} xy^2 z &= a^k a^{2j} a^{p-k-j} b^p c^{2p} \\ &= a^{p+j} b^p c^{2p} \in ADD \end{aligned}$$

   But since $j > 0, 2p + j \neq 2p$ Thus, there are more as and bs than ADD is able to accept. Since this is a contradiction to our assumption that ADD is regular, we know that it must be nonregular.

2. Suppose $POWERS$ is regular. Let p be the pumping constant for $POWERS$. Let $w = 0^{2^p}$. Then w can be broken up into substrings xyz where $|xy| \leq p$ and $0 \leq |y|$ such that $xy^i z$ exists for all i.

   Then we know that $x = 0^k, y = 0^j$, and $z = 0^{2^p-k-j}$ where $k \geq 0, j > 0$.

   Let $i = 2$. Then:

   $$\begin{aligned} xy^2 z &= 0^k 0^{2j} 0^{2^p-k-j} \\ &= 0^{2^p+j} \in POWERS \end{aligned}$$

   But since $j > 0, 2^p + j \neq 2^p$ Thus, there are more as and bs than $POWERS$ is able to accept. Since this is a contradiction to our assumption that $POWERS$ is regular, we know that it must be nonregular.

3. (a) $A \to 0A0|0A1|1A0|1A1|1$
   (b) $A \to 0A0|1A1|1|0|\epsilon$
   (c) $A \to 0B0|0B1|1B0|1B1|B$
   $B \to A|01|10$
   (d) $A \to aFbC|aS|\epsilon$
   $F \to aFb|\epsilon$
   $S \to bSc|\epsilon$
   $C \to cC|\epsilon$

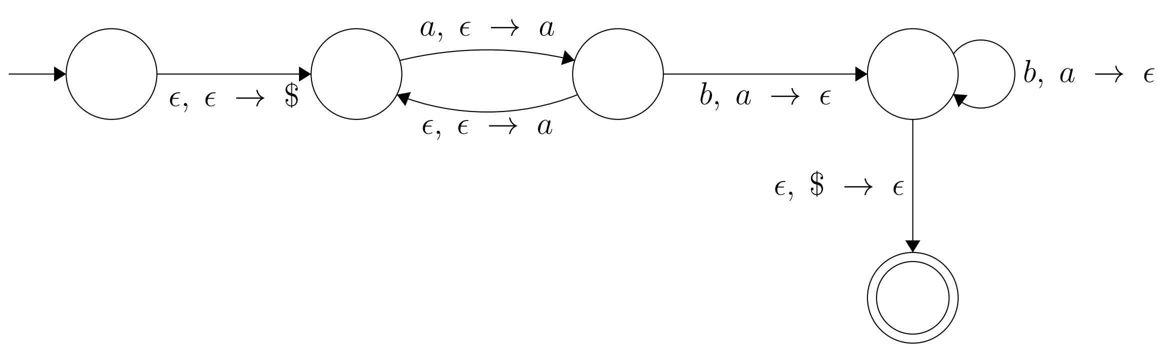
(e) $A \to BbbaB$
$B \to aB|bB|\epsilon$

(f) $A \to AaA|AbaA|AbbB|B|\epsilon$
$B \to bB|\epsilon$

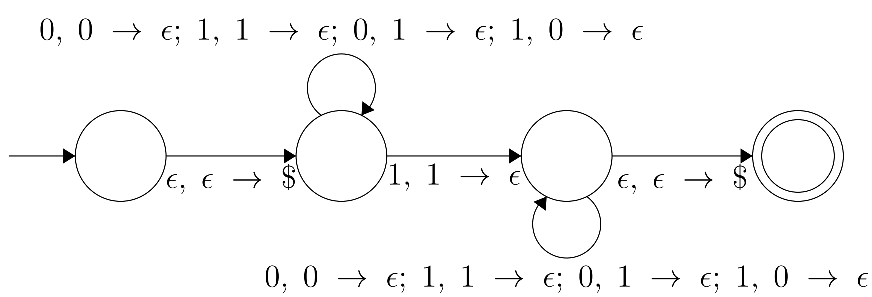
(g) $A \to aab|aaAB|a$
$B \to b|\epsilon$

(h) $S \to aSd|aDc|bAd|bMc$
$D \to aDc|bMc|\epsilon$
$A \to bMc|bAd|\epsilon$
$M \to bMc|\epsilon$

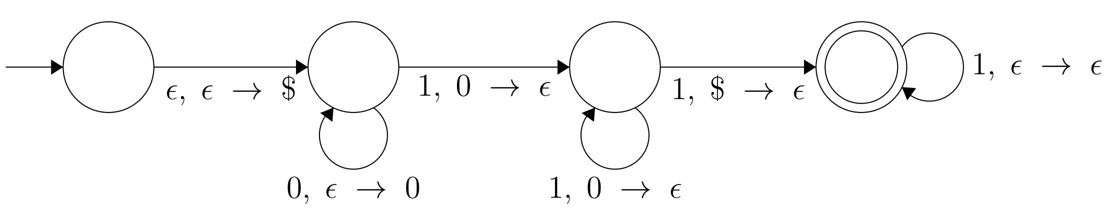
4. (a) This PDA works by pushing two as to the stack for every one it reads. Then, when it comes time to read the bs, it pops off one a for each b. If there are twice as many bs as as that were read in, it will end up in an accepting state.



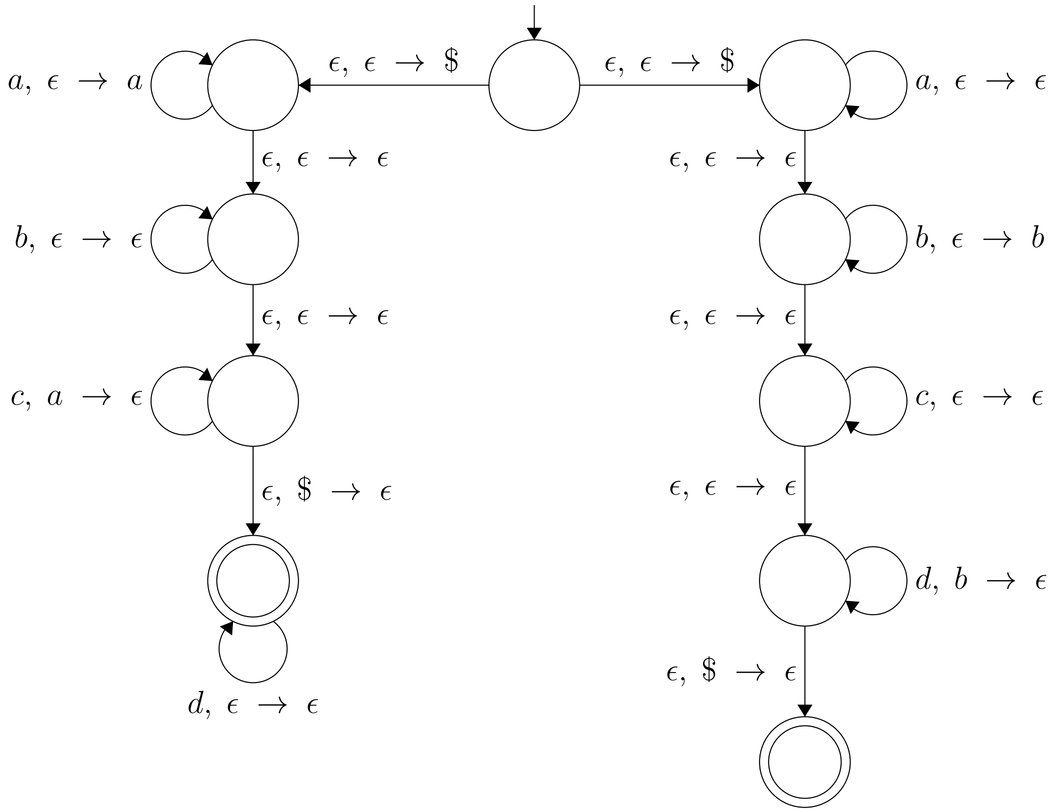
(b) This PDA will read in the first half until you have pushed a 1 and then pop off a 1 indicating that there is a 11 in the middle of the input string, and then begins popping elements off of the stack.



(c) Read 0s and fill the stack, then pop them off each time we read a 1. If we read a 1 and pop a \$ off the stack, that means that there are more 1s than 0s, so we move to an accepting state and read through the rest of the input string.



(d) This PDA will use a non-deterministic approach to split the input into one of two branches that will accept if $i = k$ on the left or $j = l$ on the right.



5. To prove that context free languages are closed under union, concatenation, and star, we will construct a set of rules such that the grammars will incorporate those properties. Let us have two context free languages S_1 and S_2. To union the two langauges, construct a new language $S_1 \cup S_2 \rightarrow S_1|S_2$. To concatenate the two languages, construct a new language $S_1 \cdot S_2 \rightarrow S_1S_2$. To star closure a language, construct a new language for $S_1^* \rightarrow S_1S_1^*|\epsilon$.