

# CS2200 Homework 6

Evan Wilcox

Due April 16, 2019

1. (a) What are the variables of  $G$ ?  
R, S, T, X
- (b) What are the terminals of  $G$ ?  
a, b
- (c) Which is the start variable of  $G$ ?  
R
- (d) Give three strings in  $L(G)$ .  
ab  
aab  
babbaabb
- (e) Give three strings not in  $L(G)$ .  
ba  
baab  
ababa
- (f) True or False:  $T \rightarrow^* aba$ .  
True
- (g) True or False:  $T \rightarrow^* aba$ .  
True
- (h) True or False:  $T \rightarrow^* T$ .  
False
- (i) True or False:  $T \rightarrow^* T$ .  
False
- (j) True or False:  $XXX \rightarrow^* aba$ .  
True
- (k) True or False:  $X \rightarrow^* aba$ .  
False
- (l) True or False:  $T \rightarrow^* XX$ .  
True
- (m) True or False:  $T \rightarrow^* XXX$ .  
True
- (n) True or False:  $S \rightarrow^* \epsilon$ .  
False
- (o) Give a description in English of  $L(G)$ .

2. (a)  $\{w \mid w \text{ contains at least three 1's}\}$ 
  - i.  $S \rightarrow P1P1P1P$
  - ii.  $A \rightarrow A0|A1|\epsilon$
- (b)  $\{w \mid w \text{ starts and ends with the same symbol}\}$ 
  - i.  $S \rightarrow 0A0|1A1|0|1$
  - ii.  $A \rightarrow 0A|1A|\epsilon$
- (c)  $\{w \mid \text{the length of } w \text{ is odd}\}$ 
  - i.  $S \rightarrow ASA|A$
  - ii.  $A \rightarrow 0|1$
- (d)  $\{w \mid \text{the length of } w \text{ is odd and its middle symbol is a 0}\}$ 
  - i.  $S \rightarrow ASA|0$
  - ii.  $A \rightarrow 0|1$
- (e)  $\{w \mid w = wR, \text{ that is, } w \text{ is a palindrome}\}$ 
  - i.  $S \rightarrow 0S0|1S1|0|1|\epsilon$
- (f) The empty set
  - i.  $S \rightarrow S$

3. The structure of the .pda file should be as follows:  
 One line beginning with an A followed by the characters in the pda's tape alphabet. One line beginning with a K followed by the characters in the pda's stack alphabet and the first character should be the end stack character. One line per state starting with a S followed by the name of the state and a boolean indicating if the state is a final state. One line starting with a B followed by the name of the starting state. One line per delta. Starting with a D followed by the state you are in, the character you see on the tape, the character you see on the stack, the character to be replaced on the stack, and state you are going to. The @ should be used for the empty string. One line per input to be tested starting with a T then the string, immediately followed by a line starting with an O for the output.

```

6  #3
7  def pdaSim(file):
8      f = open(file, "r")
9
10     states = []
11     delta = []
12     outputs = []
13     l = []
14     stack = []
15
16     for line in f:
17         l.append(line)
18
19         if line[0] == 'A':
20             alphabet = line[2:]
21
22         if line[0] == 'K':
23             stackAlphabet = line[2:]
24
25         if line[0] == 'S':
26             states.append([line[2:len(line)-4], int(line[len(line)-2:len(line)-1])])
27
28         if line[0] == 'B':
29             beginState = line[2:-1]
30
31         if line[0] == 'D':
32             startState = line[2:line.index(',')]
33             tape = line[line.index(',')+2:line.index(',', line.index(',')+1)]
34             stack = line[line.index(tape)+len(tape)+2:line.index(',', line.index(tape)+2)]
35             replace = line[line.index(stack)+len(stack)+2:line.index(',', line.index(stack)+2)]
36             endState = line[line.index(replace)+len(replace)+2:-1]
37
38             delta.append([startState, tape, stack, replace, endState])
39
40         if line[0] == 'T':
41             tape = line[2:-1]
42             state = beginState
43
44             for t in tape:
45                 for d in delta:
46                     if state == d[0] and d == d[1]:
47                         if d[2] != '@' and d[2] == stack[len(stack)-1] and d[1] == t:
48                             state = d[4]
49
50                         if d[2] == '@' and d[3] != '@':
51                             stack.append(d[3])
52
53                         elif d[2] != '@' and d[3] == '@':
54                             stack.pop()
55                             break
56
57             if [state, 1] in states:
58                 outputs.append("Accepted")
59             else:
60                 outputs.append("Rejected")
61
62     f.close()
63
64     w = open(file, 'w')
65     for line in l:
66         if line[0] == 'O':
67             line = line[:2] + outputs[0] + line[-1:]
68             outputs = outputs[1:]
69
70         w.write(line)
71
72     w.close()
73
74

```

5. (a)  $x$

Rule	Application	Result
$\text{Start} \rightarrow E$	Start	E
$E \rightarrow T$	E	T
$T \rightarrow F$	T	F
$F \rightarrow x$	F	x

$x+x$

Rule	Application	Result
$\text{Start} \rightarrow E$	Start	E
$E \rightarrow E+T$	E	$E + T$
$E \rightarrow T$	$E + T$	$T + T$
$T \rightarrow F$	$T + T$	$F + T$
$F \rightarrow x$	$F + T$	$x + T$
$T \rightarrow F$	$x + T$	$x + F$
$F \rightarrow x$	$x + F$	$x + x$

$(x \times x) + x$

Rule	Application	Result
$\text{Start} \rightarrow E$	Start	E
$E \rightarrow E + T$	E	$E + T$
$E \rightarrow T$	$E + T$	$T + T$
$T \rightarrow F$	$T + T$	$F + T$
$F \rightarrow (E)$	$F + T$	$(E) + T$
$E \rightarrow T$	$(E) + T$	$(T) + T$
$E \rightarrow E \times T$	$(T) + T$	$(T \times F) + T$
$T \rightarrow F$	$(T \times F) + T$	$(F \times F) + T$
$F \rightarrow x$	$(F \times F) + T$	$(x \times F) + T$
$F \rightarrow x$	$(x \times F) + T$	$(x \times x) + T$
$T \rightarrow F$	$(x \times x) + T$	$(x \times x) + F$
$F \rightarrow x$	$(x \times x) + F$	$(x \times x) + x$

$((x))$

Rule	Application	Result
$\text{Start} \rightarrow E$	Start	E
$E \rightarrow T$	E	T
$T \rightarrow F$	T	F
$F \rightarrow (E)$	F	$(E)$
$E \rightarrow T$	$(E)$	$(T)$
$T \rightarrow F$	$(T)$	$(F)$
$F \rightarrow (E)$	$(F)$	$((E))$
$E \rightarrow T$	$((E))$	$((T))$
$T \rightarrow F$	$((T))$	$((F))$
$F \rightarrow x$	$((F))$	$((x))$