

CS375 WK7

Jason N Mansfield

August 22, 2011

```
graph TD
    START([START]) --> RD1{READ}
    RD1 -- c --> RD2{READ}
    RD1 -- d --> RD3{READ}
    RD2 -- Δ --> ACCEPT([ACCEPT])
    RD2 -- c --> RD1
    RD2 -- d --> RD4{READ}
    RD3 -- c --> RD1
    RD3 -- d --> RD4
    RD4 -- Δ --> REJECT1([REJECT])
    RD4 -- c --> RD3
    RD4 -- d --> RD5{READ}
    RD5 -- Δ --> REJECT2([REJECT])
    RD5 -- c --> RD2
    RD5 -- d --> RD6{READ}
    RD6 -- Δ --> REJECT3([REJECT])
    RD6 -- c --> RD5
```

Figure 3: Q02: FA

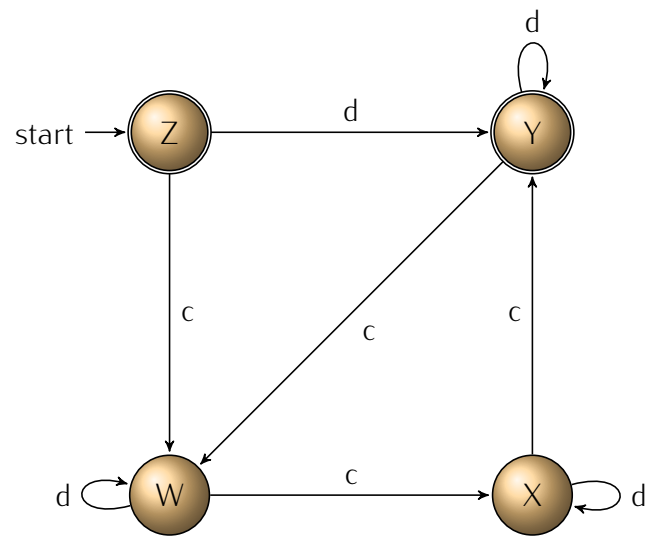


Figure 4: Q02: PDA

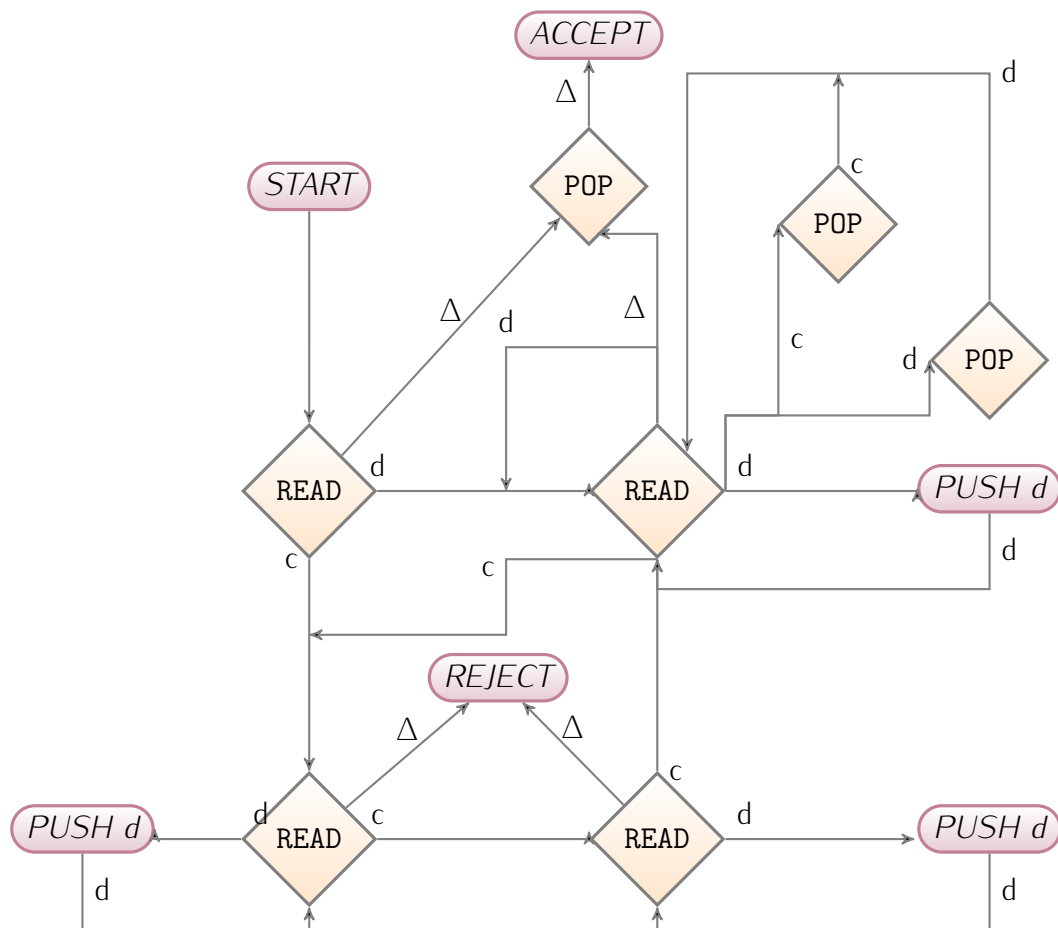


Figure 5: Q03a: Conversion Form

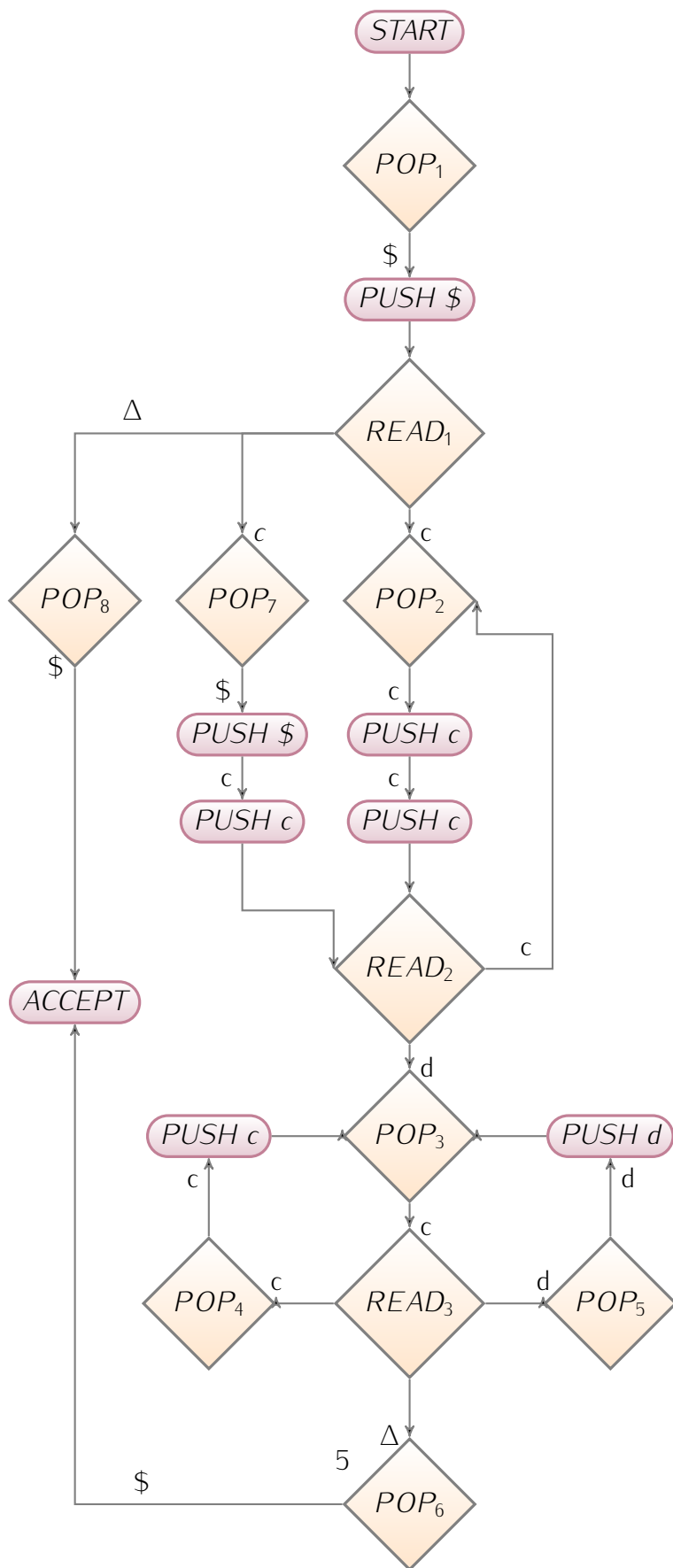


Figure 6: Q03.a Summary Table

FROM Where	To Where	READ What	POP What	PUSH What	ROW Number
START	$READ_1$	Λ	\$	\$	1
$READ_1$	ACCEPT	Λ	\$	-	2
$READ_1$	$READ_2$	c	\$	c\$	3
$READ_1$	$READ_2$	c	c	cc	4
$READ_2$	$READ_2$	c	c	cc	5
$READ_2$	$READ_3$	d	c	-	6
$READ_3$	$READ_3$	c	cc	c	7
$READ_3$	$READ_3$	d	dc	d	7
$READ_3$	ACCEPT	Λ	\$	-	8

Figure 7: Q03.a Productions

Rule 1: $S \rightarrow Net(START, ACCEPT, \$)$

Rule 2:

$Net(READ_1, ACCEPT, \$) \rightarrow Row_2$

$Net(READ_2, READ_3, \$) \rightarrow Row_6$

$Net(READ_3, ACCEPT, \$) \rightarrow Row_8$

Rule 3:

$Row_1 \rightarrow \Lambda$

$Row_2 \rightarrow \Lambda$

$Row_3 \rightarrow c$

$Row_4 \rightarrow c$

$Row_5 \rightarrow c$

$Row_6 \rightarrow d$

$Row_7 \rightarrow c$

$Row_8 \rightarrow c$

Figure 8: Q03.b

STATE	STACK	TAPE
Start	Δ	c c c a a a
$Read_1$	Δ	c c c a a a
Push	$c\Delta$	c c c a a a
$Read_2$	$c\Delta$	c c a a a
Push	$cc\Delta$	c c a a a
$Read_2$	$cc\Delta$	c c c a a a
Push	$ccc\Delta$	c c c a a a
$Read_2$	$ccc\Delta$	c c c a a a
Pop	$cc\Delta$	c c c a a a
$Read_3$	$cc\Delta$	c c c a a a
Pop	$c\Delta$	c c c a a a
$Read_3$	$c\Delta$	c c c a a a
Pop	Δ	c c c a a a
$Read_3$	Δ	c c c a a a
Pop	Δ	c c c a a a
Accept	Δ	c c c a a a

Figure 9: Q03.c

STATE	STACK	TAPE
Start	Δ	c c c a c a
$Read_1$	Δ	c c c a c a
Push	$c\Delta$	c c c a c a
$Read_2$	$c\Delta$	c c c a c a
Push	$cc\Delta$	c c c a c a
$Read_2$	$cc\Delta$	c c c a c a
Push	$ccc\Delta$	c c c a c a
$Read_2$	$ccc\Delta$	c c c a c a
Pop	$cc\Delta$	c c c a c a
$Read_3$	$cc\Delta$	c c c a c a
Pop	$c\Delta$	c c c a c a
$Read_3$	$c\Delta$	c c c a c a
Pop	Δ	c c c a c a
$Read_3$	Δ	c c c a c a
Pop	Δ	c c c a c a
Accept	Δ	c c c a c a

Figure 10: Q03.d

STATE	STACK	TAPE
Start	Δ	c c c a c c
$Read_1$	Δ	c c c a c c
Push	$c\Delta$	c c c a c c
$Read_2$	$c\Delta$	c c c a c c
Push	$cc\Delta$	c c c a c c
$Read_2$	$cc\Delta$	c c c a c c
Push	$ccc\Delta$	c c c a c c
$Read_2$	$ccc\Delta$	c c c a c c
Pop	$cc\Delta$	c c c a c c
$Read_3$	$cc\Delta$	c c c a c c
Pop	$c\Delta$	c c c a c c
$Read_3$	$c\Delta$	c c c a c c
Pop	Δ	c c c a c c
$Read_3$	Δ	c c c a c c
Pop	Δ	c c c a c c
Accept	Δ	c c c a c c

Figure 11: Q03.e

STATE	STACK	TAPE
Start	Δ	c c c c a a
$Read_1$	Δ	c c c c a a
Push	$c\Delta$	c c c c a a
$Read_2$	$c\Delta$	c c c c a a
Push	$cc\Delta$	c c c c a a
$Read_2$	$cc\Delta$	c c c c a a
Push	$ccc\Delta$	c c c c a a
$Read_2$	$ccc\Delta$	c c c c a a
Push	$cccc\Delta$	c c c c c a a
$Read_2$	$cccc\Delta$	c c c c c a a
Pop	$ccc\Delta$	c c c c a a
$Read_2$	$ccc\Delta$	c c c c a a
Pop	$cc\Delta$	c c c c a a
$Read_3$	$cc\Delta$	c c c c a a
Pop	$c\Delta$	c c c c a a
Reject	$c\Delta$	c c c c a a

Q04: a^n and b^n must be on the outside edges of the string and not larger than one. While this language is not quite an EVENPALINDROME it must be even with the internal b^m and a^m grouped in twos. This allows the string to get pumped up in the center and remain even.

Figure 12: Q05-07: Conversion Form

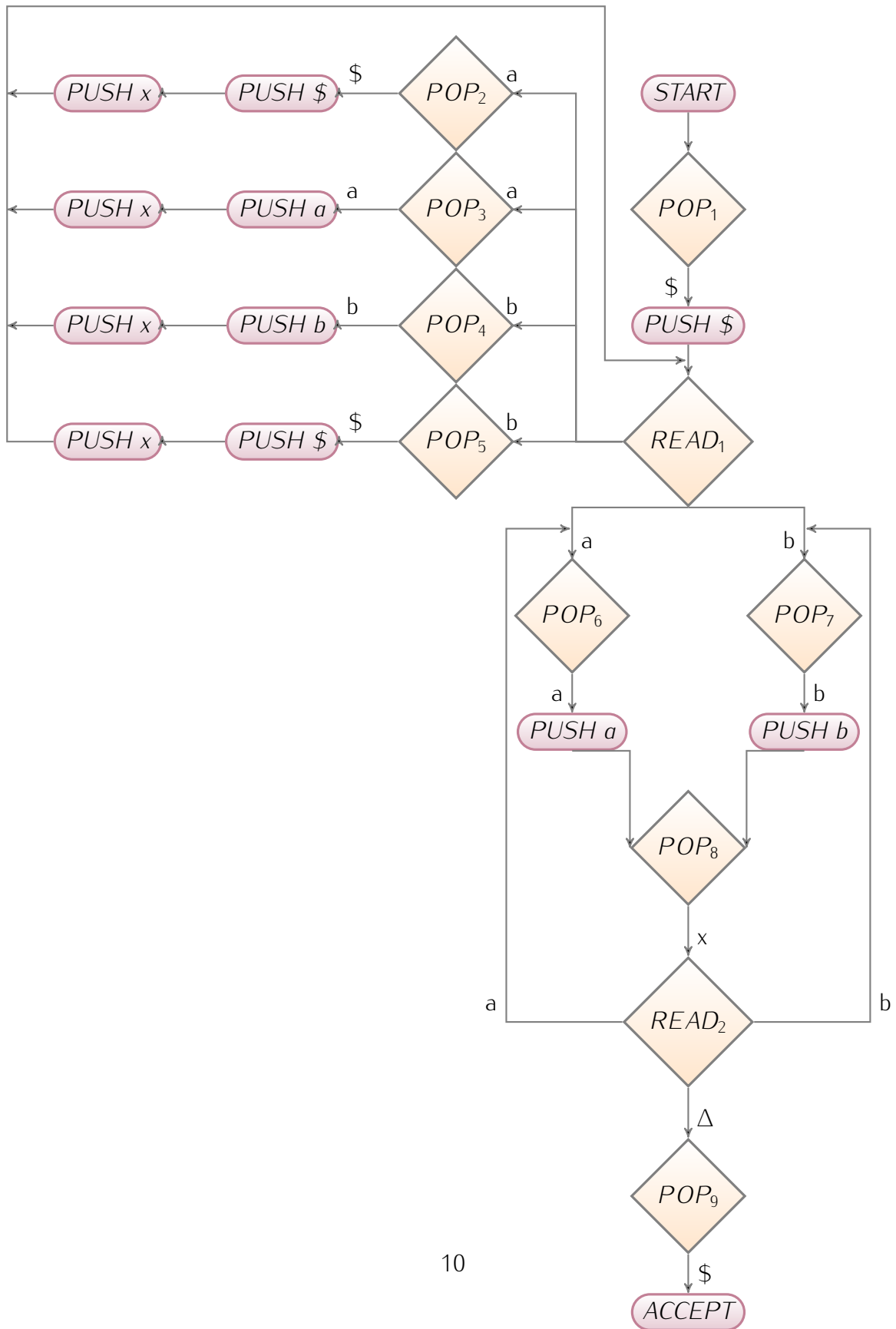


Figure 13: Q05-07: Summary Table

FROM Where	To Where	READ What	POP What	PUSH What	ROW Number
START	$READ_1$	Λ	\$	\$	1
$READ_1$	$READ_1$	a	\$	x\$	2
$READ_1$	$READ_1$	a	a	xa	3
$READ_1$	$READ_1$	b	b	xb	4
$READ_1$	$READ_1$	b	\$	x\$	5
$READ_1$	$READ_2$	a	ax	a	6
$READ_1$	$READ_2$	b	bx	b	7
$READ_2$	$READ_2$	a	ax	a	8
$READ_2$	$READ_2$	b	bx	b	9
$READ_2$	ACCEPT	Δ	\$	-	10

Figure 14: Q05

$Net(READ_1, READ_1, a)Row2$
 $Net(READ_1, READ_2, b)Row7$
 $Net(READ_2, ACCEPT, \Delta)Row10$

Example with Tape

STATE	STACK	TAPE
Start	Δ	a b
$Read_1$	Δ	a b
$Push_x$	x Δ	a b
$Read_1$	x Δ	a b
Pop	Δ	a b
$Read_2$	Δ	a b
Pop	Δ	a b
Accept	Δ	a b

Figure 15: Q06

$Net(READ_1, READ_1, b)Row5$
 $Net(READ_1, READ_1, b)Row5$
 $Net(READ_1, READ_1, b)Row5$
 $Net(READ_1, READ_2, a)Row6$
 $Net(READ_2, ACCEPT, \Delta)Row10$

Example with Tape

STATE	STACK	TAPE
Start	Δ	b b b a
$Read_1$	Δ	b b b a
$Push_x$	$x\Delta$	b b b a
$Read_1$	$x\Delta$	b b b a
$Push_x$	$xx\Delta$	b b b a
$Read_1$	$xx\Delta$	b b b a
Pop	$x\Delta$	b b b a
$Read_2$	$x\Delta$	b b b a
Pop	Δ	b b b a
$Read_2$	Δ	b b b a
Pop	Δ	b b b a
Accept	Δ	b b b a

Figure 16: Q07: FA PALINDROME

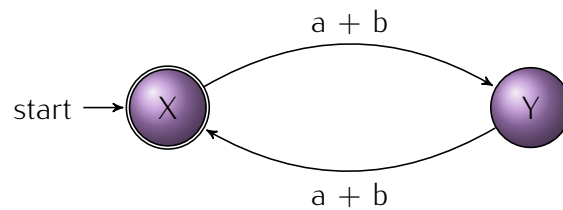


Figure 17: Q08-11: Conversion Form

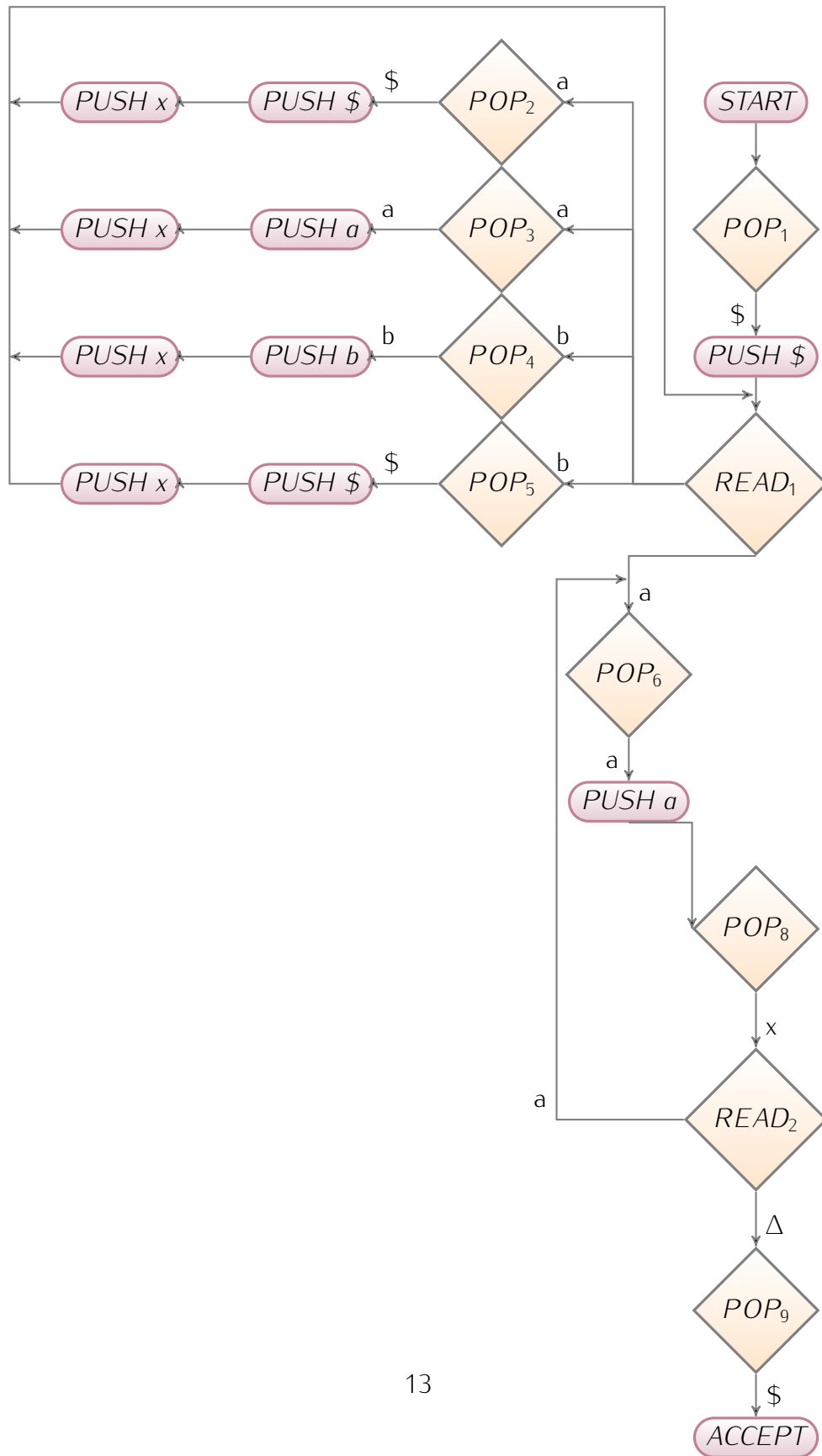


Figure 18: Q08-11: Summary Table

FROM Where	To Where	READ What	POP What	PUSH What	ROW Number
START	$READ_1$	Λ	\$	\$	1
$READ_1$	$READ_1$	a	\$	x\$	2
$READ_1$	$READ_1$	a	a	xa	3
$READ_1$	$READ_1$	b	b	xb	4
$READ_1$	$READ_1$	b	\$	x\$	5
$READ_1$	$READ_2$	a	ax	a	6
$READ_2$	$READ_2$	a	ax	a	8
$READ_2$	ACCEPT	Δ	\$	-	10

Figure 19: Q08

$Net(READ_1, READ_1, a)Row2$
 $Net(READ_1, READ_2, a)Row6$
 $Net(READ_2, ACCEPT, \Delta)Row10$

Example with Tape

STATE	STACK	TAPE
Start	Δ	a b
$Read_1$	Δ	a b
$Push$	x Δ	a b
$Read_1$	x Δ	a b
Pop	Δ	a b
$Read_2$	Δ	a b
Pop	Δ	a b
Accept	Δ	a b

Figure 20: Q09

$Net(READ_1, READ_1, b)Row5$
 $Net(READ_1, READ_1, a)Row2$
 $Net(READ_1, READ_1, b)Row5$
 $Net(READ_1, READ_2, a)Row6$
 $Net(READ_2, READ_2, a)Row8$
 $Net(READ_2, READ_2, a)Row8$
 $Net(READ_2, ACCEPT, \Delta)Row10$

Example with Tape

STATE	STACK	TAPE
Start	Δ	b a b a a a
$Read_1$	Δ	a a b a a a
$Push$	$x\Delta$	a a b a a a
$Read_1$	$x\Delta$	a a b a a a
$Push$	$xx\Delta$	a a b a a a
$Read_1$	$xx\Delta$	a a a a a a
$Push$	$xxx\Delta$	a a a a a a
$Read_1$	$xxx\Delta$	a a a a a a
Pop	$xx\Delta$	a a a a a a
$Read_2$	$xx\Delta$	a a a a a a
Pop	$x\Delta$	a a a a a a
$Read_2$	$x\Delta$	a a a a a a b
Pop	Δ	a a a a a a b
$Read_2$	Δ	a a a a a a b
Pop	Δ	a a a a a a b
Accept	Δ	a a a a a a b

Figure 21: Q10

The PDA for 8-11 is an EVENPALINDROME therefore it is impossible to balance the amount of readable tape with the stack if not even as shown below. If instead of the below example we were to remain in READ 1 an additional cycle then we would have too many x's on the Stack and the tape would complete too soon. In this case the tape completes too late.

$Net(READ_1, READ_1, b)$ Row 5
 $Net(READ_1, READ_1, a)$ Row 2
 $Net(READ_1, READ_1, b)$ Row 5
 $Net(READ_1, READ_2, a)$ Row 6
 $Net(READ_2, READ_2, a)$ Row 8
 $Net(READ_2, READ_2, b)$ Row 8
 CRASH

Example with Tape

STATE	STACK	TAPE
Start	Δ	b a b a a b
$Read_1$	Δ	a a b a a b
$Push$	$x\Delta$	a a b a a b
$Read_1$	$x\Delta$	a a b a a b
$Push$	$xx\Delta$	a a b a a b
$Read_1$	$xx\Delta$	a a a a b
$Push$	$xxx\Delta$	a a a a a b
$Read_1$	$xxx\Delta$	a a a a a b
Pop	$xx\Delta$	a a a a a b
$Read_2$	$xx\Delta$	a a a a a b
Pop	$x\Delta$	a a a a a b
$Read_2$	$x\Delta$	a a a a a b
Pop	Δ	a a a a a b
$Read_2$	Δ	a a a a a b
Pop	Δ	a a a a a b

Figure 22: Q11

Once again the PDA for 8-11 is an EVENPALINDROME therefore it is impossible to balance the amount of readable tape with the stack if not even as shown below. If instead of the below example we were to remain in READ 1 an additional cycle then we would have too many x's on the Stack and the tape would complete too soon. In this case the tape completes too late. This example is identical to the failure shown above with the exception of the extra character on the string being a 'a'

$Net(READ_1, READ_1, b)Row5$
 $Net(READ_1, READ_1, a)Row2$
 $Net(READ_1, READ_1, b)Row5$
 $Net(READ_1, READ_2, a)Row6$
 $Net(READ_2, READ_2, a)Row8$
 $Net(READ_2, READ_2, a)Row8$
 CRASH

Example with Tape

STATE	STACK	TAPE
Start	Δ	b a b a a a a
$Read_1$	Δ	a b a a a a a
Push	x Δ	a a b a a a a
$Read_1$	x Δ	a a b a a a a
Push	xx Δ	a a b a a a a
$Read_1$	xx Δ	a a a a a a a
Push	xxx Δ	a a a a a a a
$Read_1$	xxx Δ	a a a a a a a
Pop	xx Δ	a a a a a a a
$Read_2$	xx Δ	a a a a a a a
Pop	x Δ	a a a a a a a
$Read_2$	x Δ	a a a a a a a
Pop	Δ	a a a a a a a
$Read_2$	Δ	a a a a a a b
Pop	Δ	a a a a a a b