Chris Fenton
CNC - formallang
Assignment 3

Chapter 8

3. Construct a Turing machine with input alphabet {a, b} to perform each of the following operations. Note that the tape head is scanning position zero in state q_f whenever a computation terminates.

(b) Concatenate a copy of the reversed input string to the input. Input configuration q_0BuB, result q_fBuu^{R}B.

```
; Concatenate a copy of the reversed input string to the input.
; Input configuration q0BuB, result qfBuuRB.
; '<current state> <current symbol> <new symbol> <direction> <new state>'

; Algorithm:
; convert letter to uppercase and copy to the right
; go back to input char in original tape, check for done
; if not done, keep copying
; if done, cleanup

; start state
0 a * r 0
0 b * r 0
0 _ * l ci

; copy initial tape
ci a A r mra
ci b B r mrb
ci _ * r cp

; move right to paste a
mra a * r mra
mra b * r mra
mra A * r mra
mra B * r mra
mra _ * r pa

; move right to paste b
mrb a * r mrb
mrb b * r mrb
mrb A * r mrb
mrb B * r mrb
mrb _ * r pb
```

```
; paste a
pa a * r pa
pa b * r pa
pa _ a l mlc


; move left to copy
mlc _ * l mlc
mlc A * l mlc
mlc B * l mlc
mlc a * l cfd
mlc b * l cfd

; paste b
pb a * r pb
pb b * r pb
pb _ b l mlc

; check for done
cfd _ * l cfdd
cfdd A * * cl
cfdd B * * cl
cfdd _ * r cfddd
cfddd _ * r cfddd
cfddd a * * ci
cfddd b * * ci

; cleanup
cl A _ r cla
cl B _ r clb
cl _ * r finish

; cleanup paste a
cla _ a l sl

; cleanup paste b
clb _ b l sl

; shift left
sl * * l cl

; shift right
finish * * r halt-accept
```

5. Construct a Turing machine with input alphabet {a, b} to accept each of the following languages by final state.

(b) { $a^i b^j a^i b^j$ | i,j > 0 }

```
; Accept L = { a^i b^j a^i b^j | i,j > 0 }
; Example strings = {"abbabb", "aabaab", "aaabbbaaabbb"}
; '<current state> <current symbol> <new symbol> <direction> <new state>'

; Algorithm:
; Check first group of a's
; - see an a, mark it, move right till b's, move right till next a, mark it
; - - if you don't find the a, and come to a blank - fail
; - move to beginning
; Check first group of b's
; - move right till first b, mark it, move right till first A, move right till next b,
mark it
; - - if you don't find the b, and come to a blank - fail
; Move to beginning
; Convert char to lowercase
; - if anything is already lowercase, fail
; Reach end of tape
; Goto beginning of tape
; Halt

; Initial state
0 * * * cfa

; Check First A's (cfa)
cfa a A r mrb
cfa A * r cfa
cfa b * * cfb
mrb a * r mrb
mrb b * r mra
mra b * r mra
mra A * r mra
mra a A r mre
mre b * r meb
mre a * * gta
mre _ * * gta
meb b * r meb
meb _ * l gta
meb a * * halt
mra _ * * halt

; Goto Beginning after a's (gta)
gta a * l gta
gta b * l gta
gta A * l gta
gta B * l gta
gta _ * r cfa

; Check B's (cfb)
cfb b B r mrr
cfb A * r cfb
cfb a * * halt
cfb B * r cfb
cfb _ * l gtc
mrr a * r mrr
mrr A * r mrr
mrr B * r mrr
mrr b B l gtb
mrr _ * * halt

; Goto Beginning after b's (gtb)
gtb a * l gtb
gtb b * l gtb
gtb A * l gtb
gtb B * l gtb
gtb _ * r cfb

; Goto Beginning to clean (gtc)
gtc A a l gtc
gtc B b l gtc
gtc _ * r halt-accept
```

24.

25.