J2_Unid_Examples

September 11, 2019

1 Imports

[1]:
```
import sys
sys.path[0:0] = ['../..','../../3rdparty','../../..','../../../3rdparty','../../
 ↪../..','../../../../3rdparty']
from jove.DotBashers import *
from jove.Def_md2mc  import *
from jove.Def_DFA    import *
from jove.LangDef    import *  # for testing DFA actions

from jove.Def_RE2NFA   import *
from jove.Def_NFA   import *
from jove.Def_NFA2RE    import *
```

```
You may use any of these help commands:
help(ResetStNum)
help(NxtStateStr)

You may use any of these help commands:
help(md2mc)
.. and if you want to dig more, then ..
help(default_line_attr)
help(length_ok_input_items)
help(union_line_attr_list_fld)
help(extend_rsltdict)
help(form_delta)
help(get_machine_components)

You may use any of these help commands:
help(mkp_dfa)
help(mk_dfa)
help(totalize_dfa)
help(addtosigma_delta)
help(step_dfa)
help(run_dfa)
help(accepts_dfa)
```

```
help(comp_dfa)
help(union_dfa)
help(intersect_dfa)
help(pruneUnreach)
help(iso_dfa)
help(langeq_dfa)
help(same_status)
help(h_langeq_dfa)
help(fixptDist)
help(min_dfa)
help(pairFR)
help(state_combos)
help(sepFinNonFin)
help(bash_eql_classes)
help(listminus)
help(bash_1)
help(mk_rep_eqc)
help(F_of)
help(rep_of_s)
help(q0_of)
help(Delta_of)
help(mk_state_eqc_name)

You may use any of these help commands:
help(lphi)
help(lunit)
help(lcat)
help(lexp)
help(lunion)
help(lstar)
help(srev)
help(lrev)
help(shomo)
help(lhomo)
help(powset)
help(lint)
help(lsymdiff)
help(lminus)
help(lissubset)
help(lissuperset)
help(lcomplem)
help(product)
help(nthnumeric)

You may use any of these help commands:
help(mk_nfa)
help(totalize_nfa)
help(step_nfa)
```

```
help(run_nfa)
help(ec_step_nfa)
help(Eclosure)
help(Echelp)
help(accepts_nfa)
help(nfa2dfa)
help(n2d)
help(inSets)
help(rev_dfa)
help(min_dfa_brz)

You may use any of these help commands:
help(re2nfa)

You may use any of these help commands:
help(RE2Str)
help(mk_gnfa)
help(mk_gnfa_from_D)
help(dfa2nfa)
help(del_gnfa_states)
help(gnfa_w_REStr)
help(del_one_gnfa_state)
help(Edges_Exist_Via)
help(choose_state_to_del)
help(form_alt_RE)
help(form_concat_RE)
help(form_kleene_RE)
```

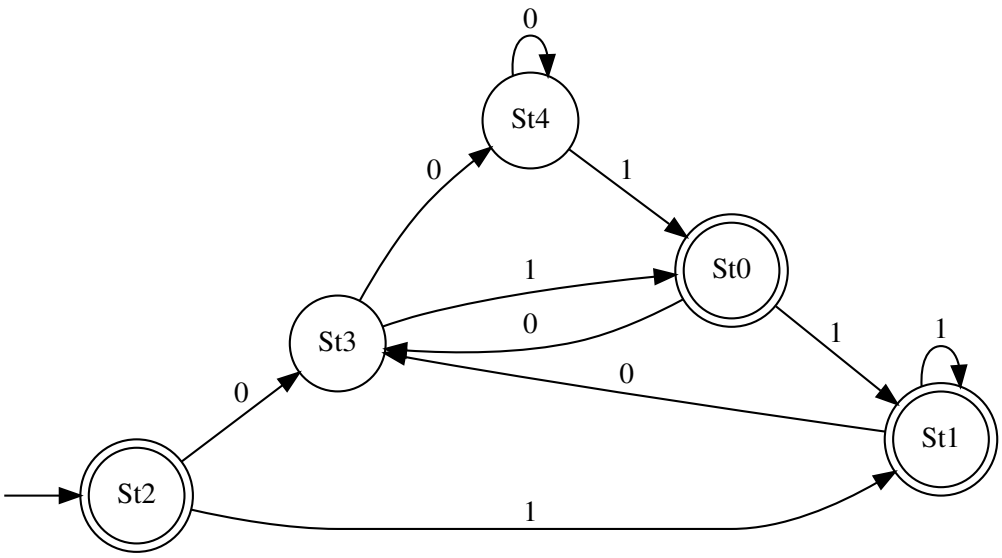
2 Question 2(c)

[2]:
```
DFA1 = nfa2dfa(re2nfa("(00*1+1)*"))
```

```
Generating LALR tables
```

[3]:
```
dotObj_dfa(DFA1)
```

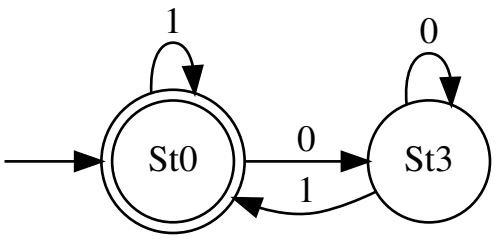
[3]:



3 Question 2(e)

[4]:
```
dotObj_dfa(min_dfa(DFA1))
```

[4]:

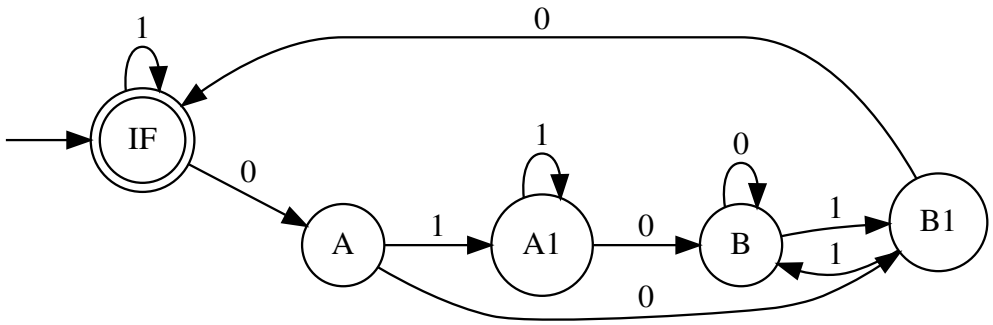


4 Question 3(a)

[5]:
```
DBloated = md2mc('''DFA
IF : 0 -> A
A : 0 -> B1
B : 0 -> B
IF: 1 -> IF
A : 1 -> A1
B : 1 -> B1
A1: 0 -> B
A1 : 1 -> A1
```

```
B1 : 0 -> IF
B1 : 1 -> B
''')
dotObj_dfa(DBloated)
```

[5]:



[6]:
```
Gbloat = mk_gnfa_from_D(DBloated)
```

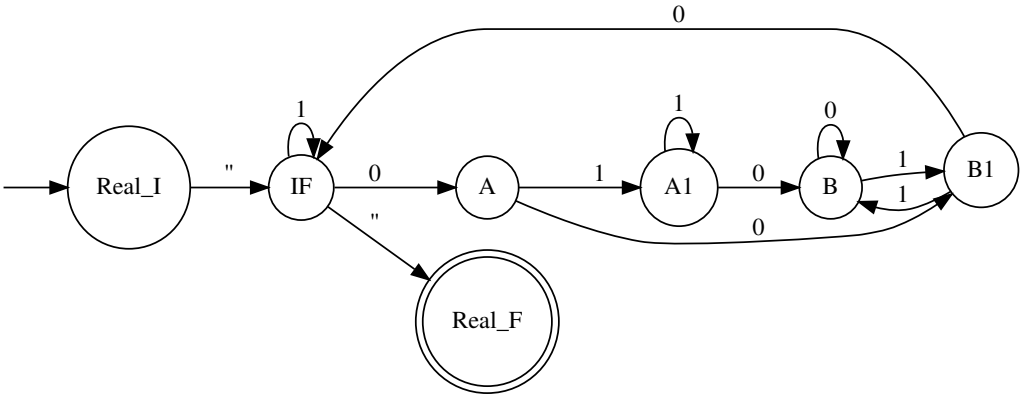
5 Question 3(c) and 3(d)

[7]:
```
(Gfinal, DO, RE) = del_gnfa_states(Gbloat, ['A1','A','B','B1','IF'])
```

```
**** Eliminating state A1 ****
**** Eliminating state A ****
**** Eliminating state B ****
**** Eliminating state B1 ****
**** Eliminating state IF ****
```

[8]:
```
DO[0]
```

[8]:

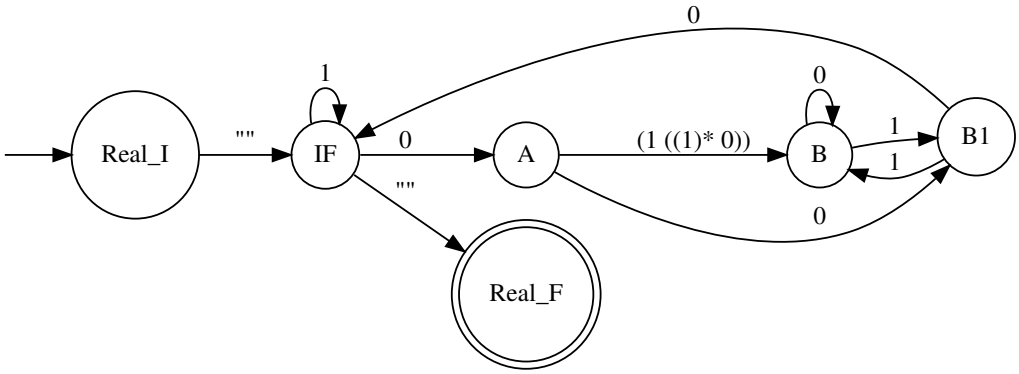


[9]:
```
DO[0].render('fig1')
```

[9]: 'fig1.pdf'

[10]:
```
DO[1]
```

[10]:

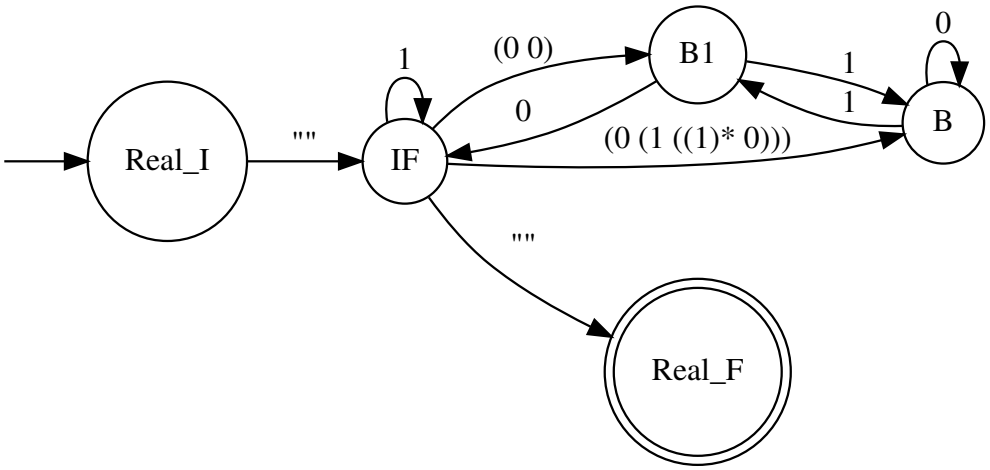


[11]:
```
DO[1].render('fig2')
```

[11]: 'fig2.pdf'

[12]:
```
DO[2]
```

[12]:

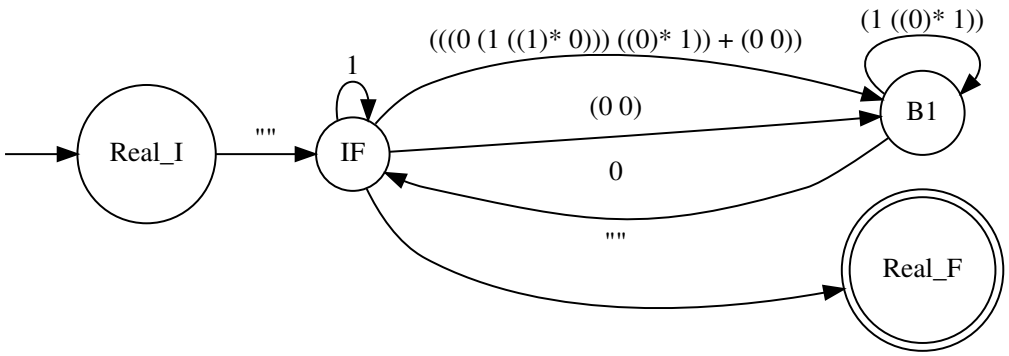


[13]:
```
DO[2].render('fig3')
```

[13]: 'fig3.pdf'

[14]:
```
DO[3]
```

[14]:

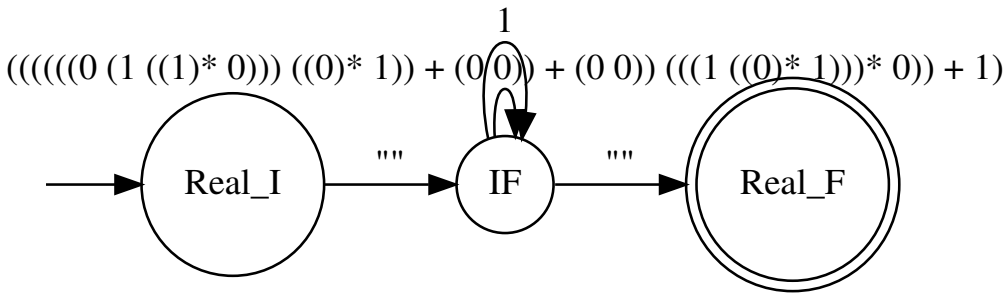


[15]:
```
DO[3].render('fig4')
```

[15]:
```
'fig4.pdf'
```

[16]:
```
DO[4]
```

[16]:

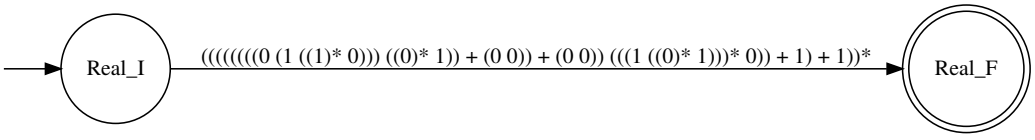


[17]:
```
DO[4].render('fig5')
```

[17]:
```
'fig5.pdf'
```

[18]:
```
DO[5]
```

[18]:



[19]:
```
DO[5].render('fig6')
```

[19]:
```
'fig6.pdf'
```

6 Question 3(e)

[20]:
```
RE
```

[20]:
```
'(((((((((0 (1 ((1)* 0))) ((0)* 1)) + (0 0)) + (0 0)) (((1 ((0)* 1)))* 0)) + 1) +
1))*'
```

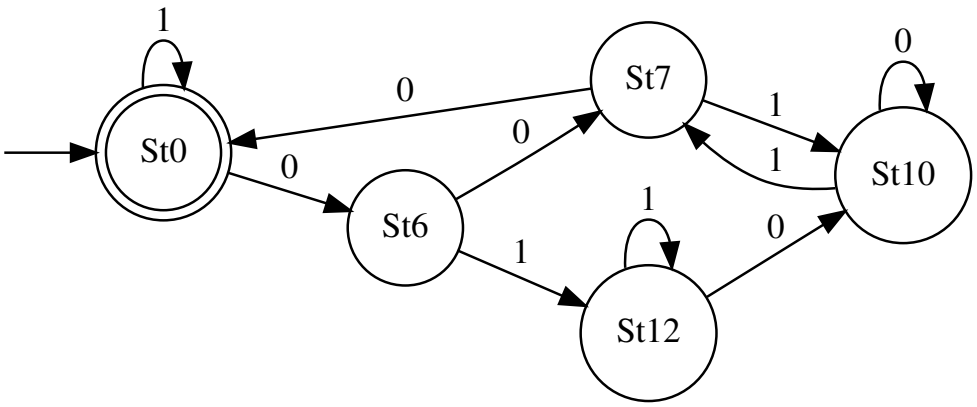
[21]:
```
RE
```

[21]:
```
'(((((((((0 (1 ((1)* 0))) ((0)* 1)) + (0 0)) + (0 0)) (((1 ((0)* 1)))* 0)) + 1) +
1))*'
```

[22]:
```
dotObj_dfa(min_dfa(nfa2dfa(re2nfa(RE))))
```

```
Generating LALR tables
```

[22]:

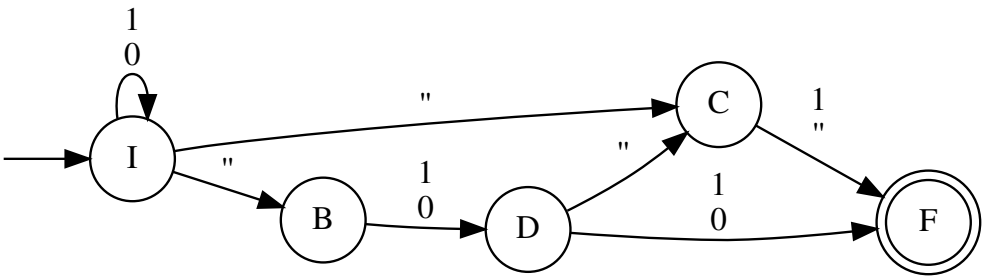


7 Question 4(b)

[23]:
```
N5 = md2mc('''NFA
I : 0|1 -> I
I : ''  -> B,C
B : 0|1 -> D
D : 0|1 -> F
D : ''  -> C
C : 1|''-> F
''')
```

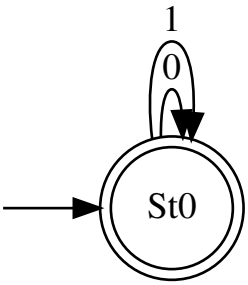
[24]:
```
dotObj_nfa(N5, FuseEdges=True)
```

[24]:



[25]:
```
JD5 = dotObj_dfa(min_dfa(nfa2dfa(N5)))
JD5
```

[25]:



8 End