In [1]:
```
from automata.notebook import *
check_answer = lambda dfa: check_dfa_language_from_words(dfa, 'aaaaaaab bbbbbba aaaaab ba bbbbba aab aaaaaab aaaab aaab bbbba ab
```

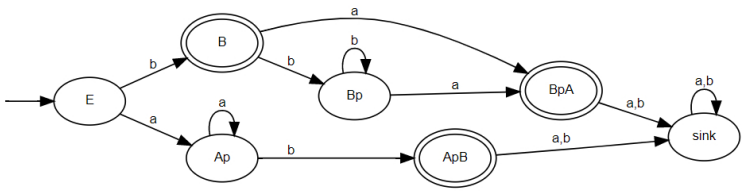
Enter a DFA for the language $\{ a^n b \mid n \geqslant 0 \} \cup \{ b^n a \mid n \geqslant 1 \}$, i.e. the language of strings of a number of a's followed by a b or a positive number of b's followed by an a.

In [19]:
```
dfa = '''
input_symbols a b
states E Ap ApB B BpA Bp sink
initial E
final BpA ApB B
E Ap a
E B b
B Bp b
B BpA a
Ap Ap a
Ap ApB b
Bp Bp b
Bp BpA a
ApB sink a
ApB sink b
BpA sink a
BpA sink b
sink sink a
sink sink b


'''
```

In [20]:
```
show(dfa)
```

Out[20]:



In [21]:
```
check_answer(dfa)
```

OK

If the DFA above is not OK, inspect its execution for a selected word, e.g. b.

In [22]:
```
simulate_dfa(dfa,'b')
```

State	Word
E	b
B	

In [ ]: