CS442/542

DFA Minimization

DFA Minimization

- Minimize the DFA $(D, \boldsymbol{\Sigma}, \delta, d_0, D_A\}$
- The algorithm builds a new machine from subsets of states of the original machine
- The algorithm first builds two subsets: the set of final states and the set of non-final states
- A subset is split if the subset has a conflict on a symbol
- A subset has a conflict on a symbol, c, when the transitions on c of two (or more) states in the subset do not go to states in the same subset.
- The algorithm halts when no subsets have conflicts (i.e. no more splits need to be done)

Split (S is a set of states from the original DFA)

```
Split(S) {
    for each c ∈ Σ do
        if c splits S into s1 and s2
            then return {s1, s2};
    end;
    return S;
}
```

DFA Minimization

```
T = { D_A, (D – D_A) };
P = ∅
while ( P ≠ T ) do
    P  ←  T ;
    T  ←  ∅;
    for each p ∈ P do
        T = T ∪ Split(p);
    end;
end;
```

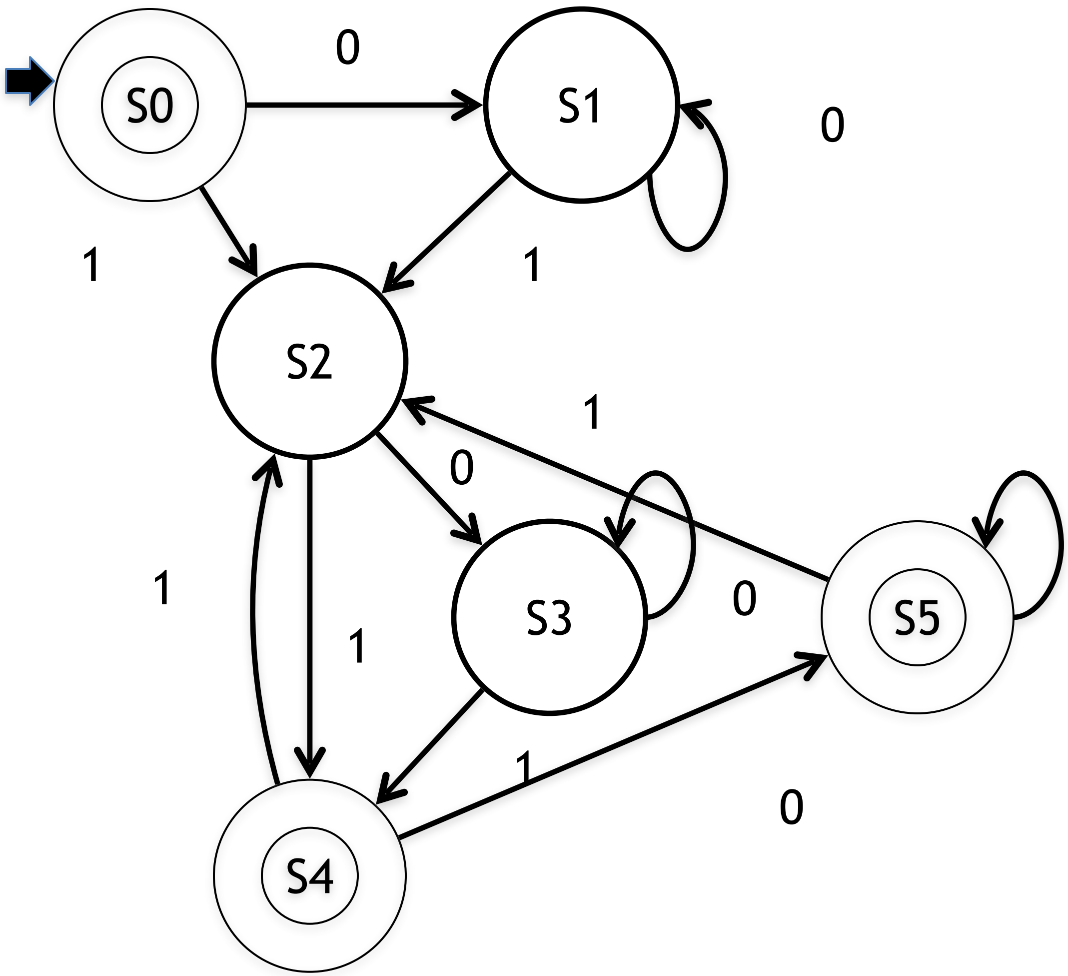
DFA Minimization

- To build a minimized DFA from a DFA create a new DFA consisting of maximal consistent states which are subsets of states of the original DFA. These subsets form the set of states in the minimized DFA
- Consistent State
  - Suppose S is a set of states s1, s2,…Sn
  - Suppose δ is the transition function of the original DFA
  - A set of states, S, is consistent if for each a in Σ and for each si in S, δ(si,a) = qi where all the qi belong to the same consistent state
- A maximal consistent state is a state that is consistent and to which a new state cannot be added and the state remains consistent

DFA Minimization

- To produce a set of maximal consistent states do the follow
  - Create two subsets of states: final and non-final states
  - For each a in Σ if for some S all the transitions on a do not lead to the same subset of states, create subsets of S that lead to the same subset of states.
  - Continue the process until for each S in the new machine and for each a in Σ the transitions on a lead to the same subset

DFA for (0*10*10*)*



DFA Minimization

- Round 1

	S0, S4, S5	S1, S2, S3
0	S1, S5, S5	S1, S3, S3

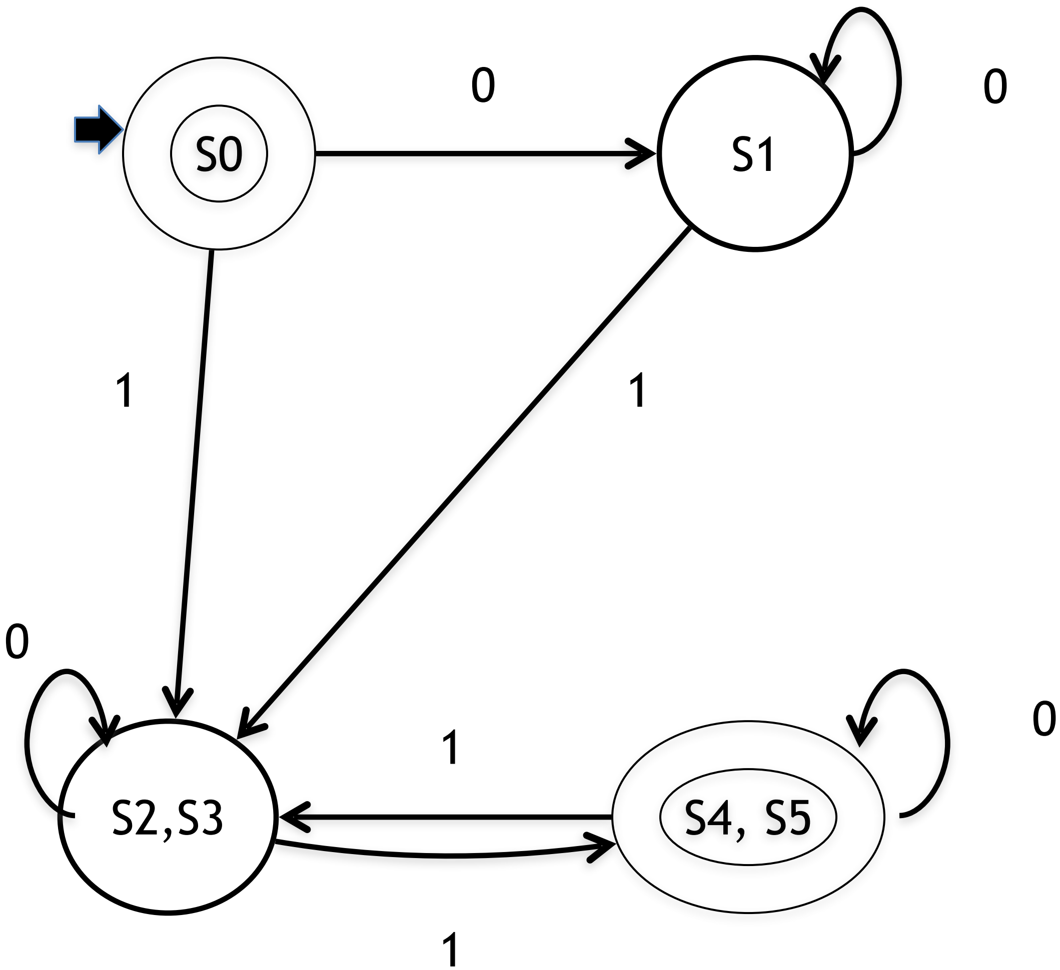
- Round 2

	S0	S4, S5	S1, S2, S3
1	S2	S2, S2	S2, S4, S4

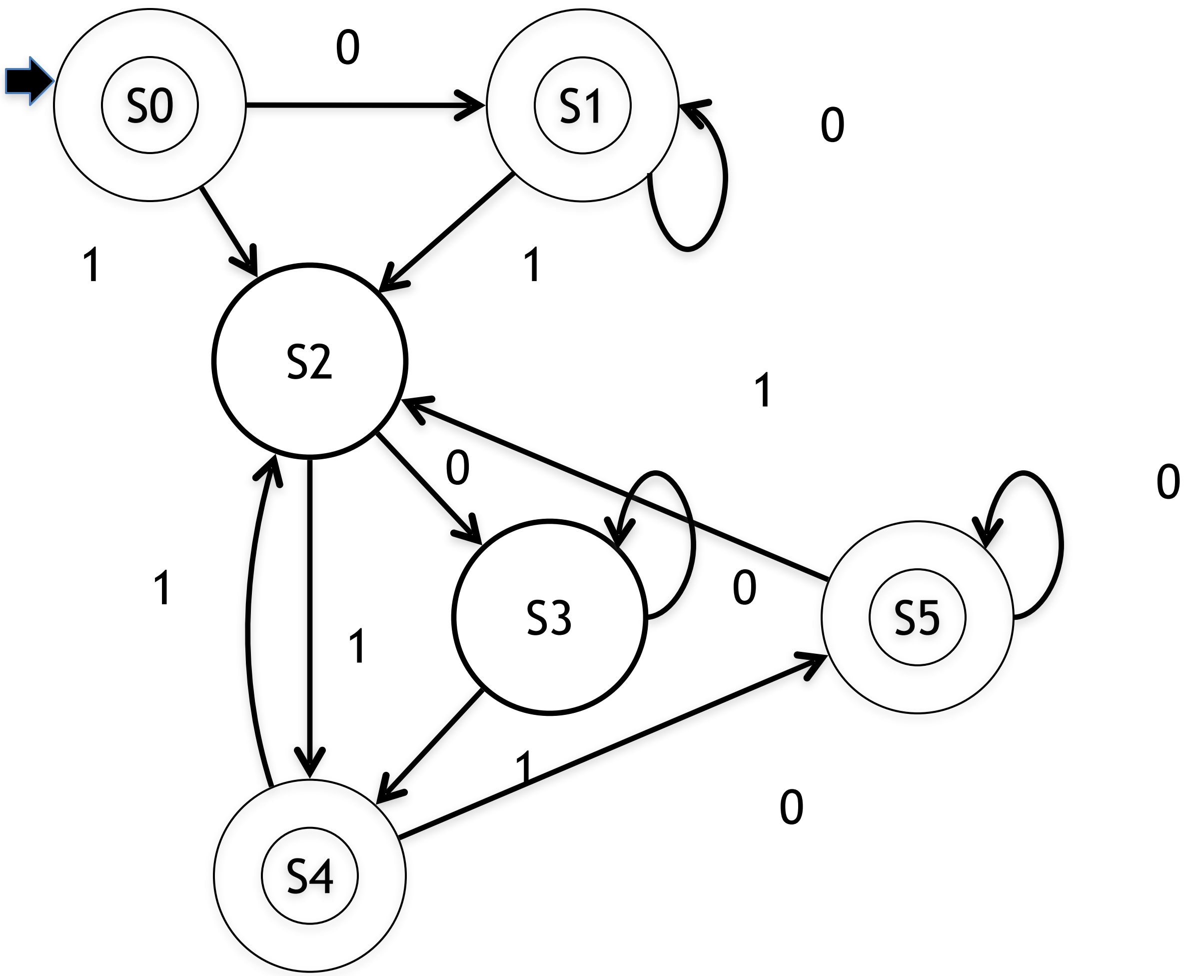
- Round 3

	S0	S4, S5	S1	S2, S3
0	S1	S5, S5	S1	S3, S3
1	S2	S2, S2	S2	S4, S4

DFA Minimization



DFA for 0*(10*10*)*

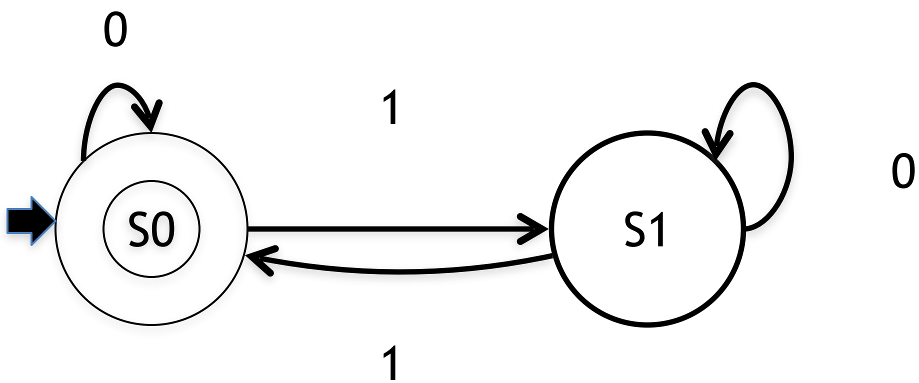


DFA Minimization

- Round 1

	S0, S1, S4, S5	S2, S3
0	S1, S1, S5, S5	S3, S3
1	S2, S2, S2, S2	S4, S4

DFA Minimization



DFA Minimization

- Potential Problems
  - The transition function is not total
  - Dead states