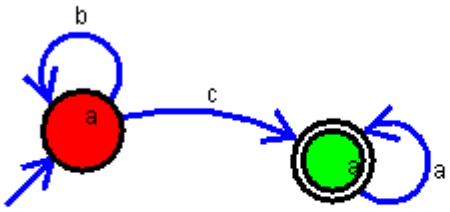
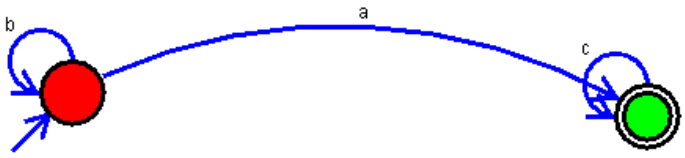
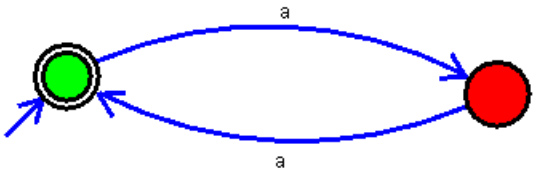
1. Design DFA to accept bcaaaaaaaaaaaaa, bc, and c



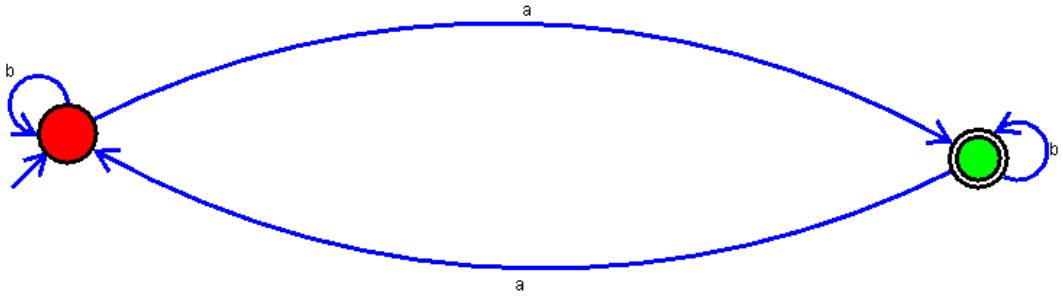
2. Design DFA using simulator to accept the input string “a” ,”ac”,and ”bac”.



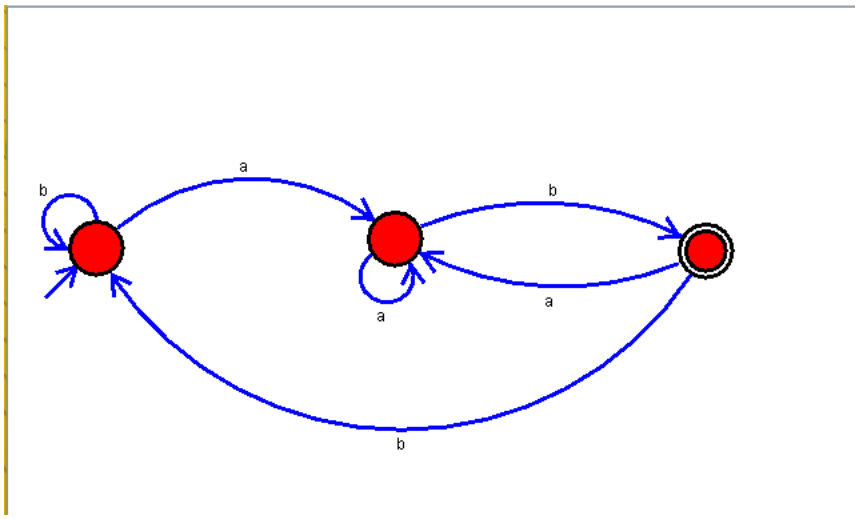
3. Design DFA using simulator to accept even number of a’s.



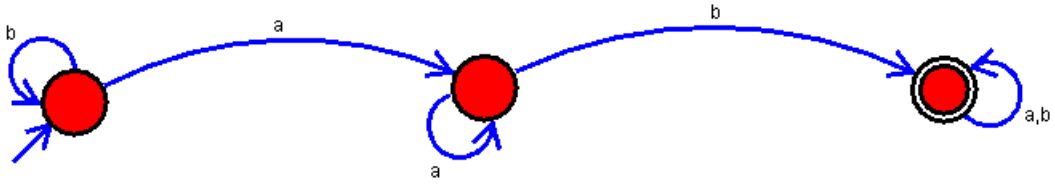
4. Design DFA using simulator to accept odd number of a’s



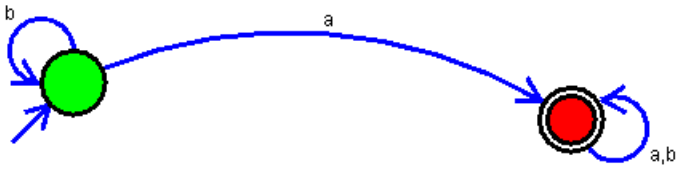
5. Design DFA using simulator to accept the string the end with ab over set {a,b}
W= aaabab



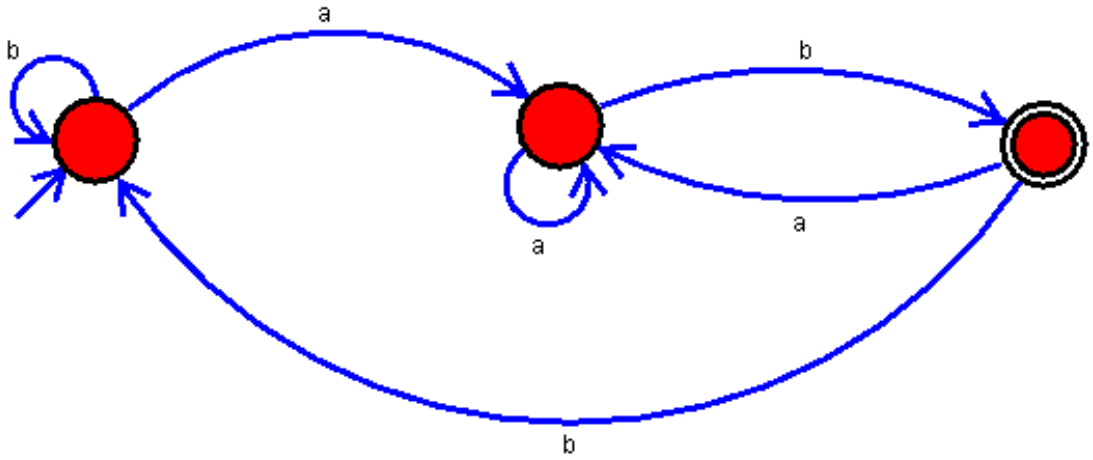
6. Design DFA using simulator to accept the string having 'ab' as substring over the set {a,b}



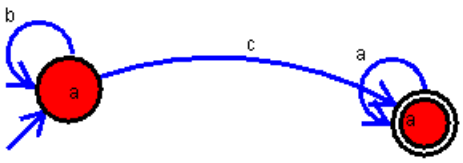
7. Design DFA using simulator to accept the string start with a or b over the set {a,b}



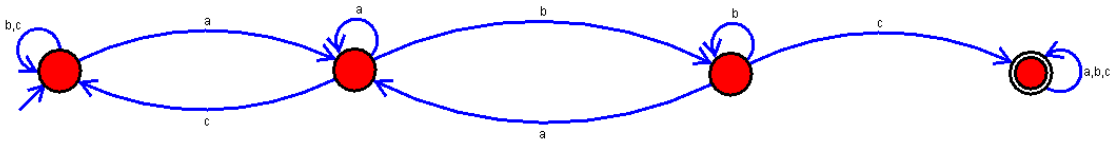
8. Design DFA using simulator to accept the string the end with ab over set {a,b}
W= abbaabab



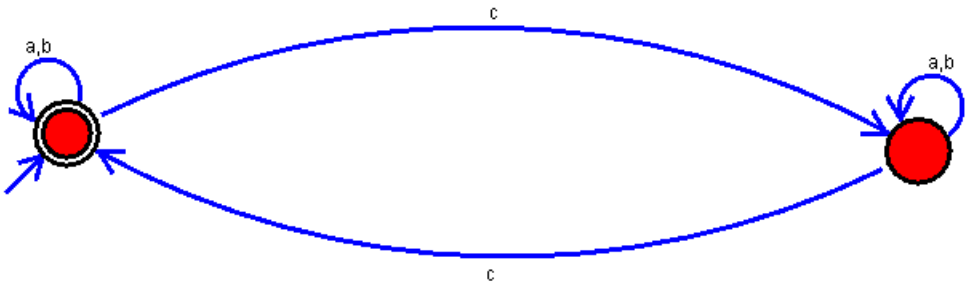
9. Design DFA using simulator to accept the input string “bc” ,”c”,and ”bcaaa”.



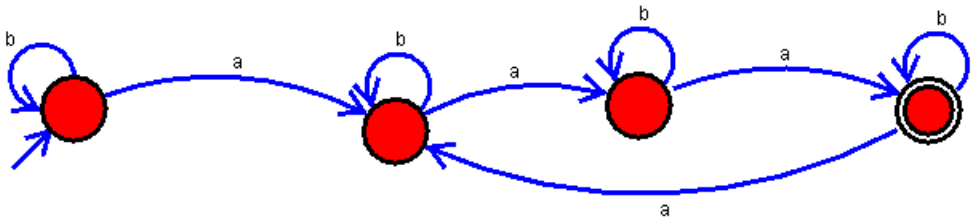
10. Design DFA using simulator to accept the string having ‘abc’ as substring over the set {a,b,c}



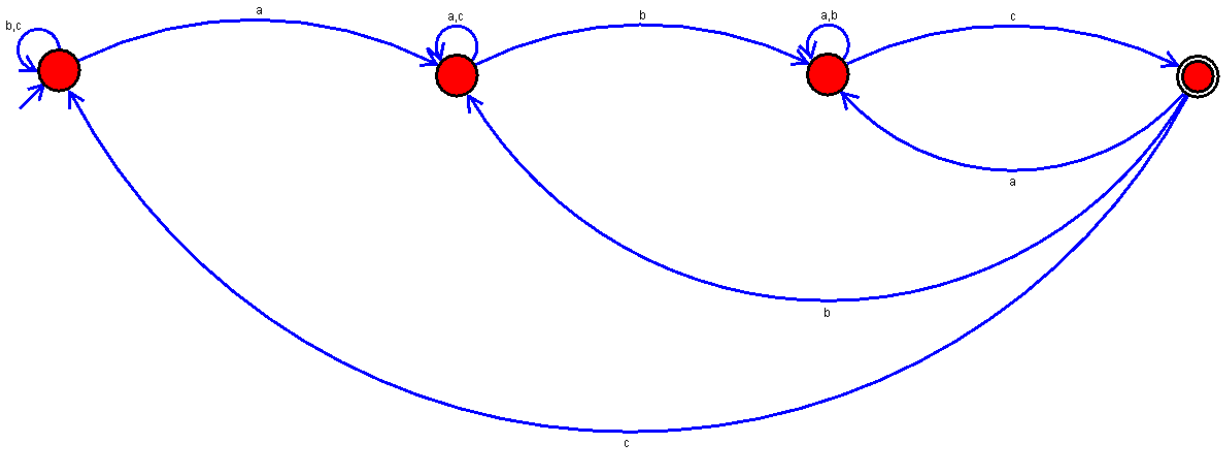
11. Design DFA using simulator to accept even number of c’s over the set {a,b,c}



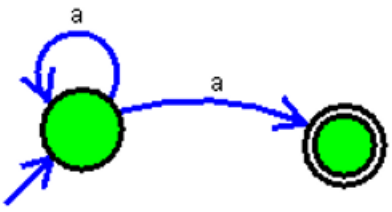
12. Design DFA using simulator to accept strings in which a's always appear tripled over input {a,b}



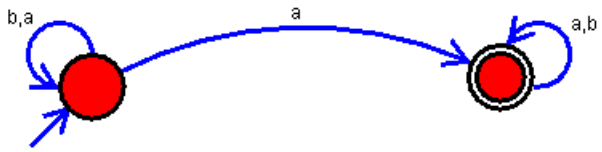
13. Design DFA using simulator to accept the string the end with abc over set {a,b,c)
W= abbaababc



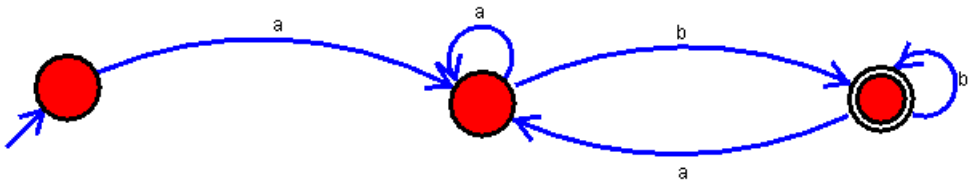
14. Design NFA to accept aaaaaa



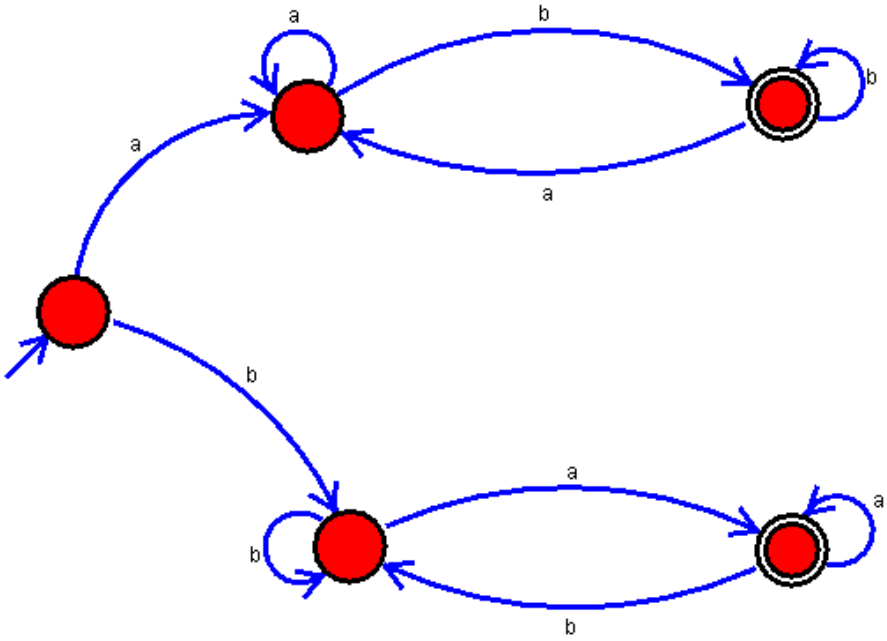
15. Design NFA to accept any number of a's where input={a,b}.



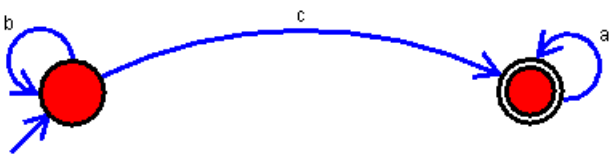
16. Design NFA using simulator to accept the string the start with a and end with b over set {a,b} and check W= abaab is accepted or not



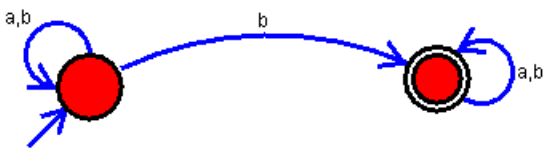
17. Design NFA using simulator to accept the string that start and end with different symbols over the input {a,b}.



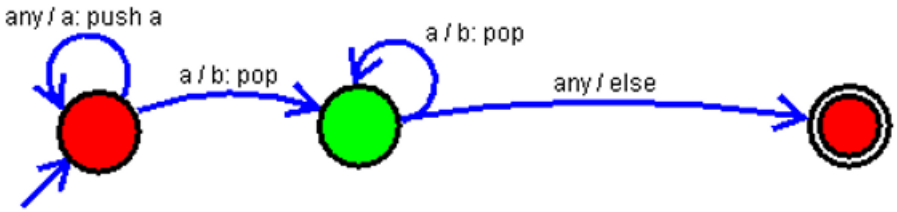
18. Design NFA using simulator to accept the input string “bbc” ,”c”,and ”bcaaa”.



19. Design NFA to accept any number of b's where input={a,b}.



20. Design PDA using simulator to accept the input string $a^n b^n$



21. Design PDA using simulator to accept the input string $a^n b^{2n}$