The transition table of the using names associated above is:

S	0	1
A	A	B
B	C	D
C	E	A
D	A	B
E	F	G
F	H	I
G	J	E
H	F	G
I*	J	E
J	K	H
K*	A	B

6. Regular sets are closed under prefix.

2.9. EXAMPLES OF CONSTRUCTION OF FINITE AUTOMATA

1. Construct A finite automata accepting the set of all strings of 0's and 1's, with at most one pair of consecutive 0's and at most one pair of consecutive 1's.

   A transition diagram of finite automata accepting the set of all strings of 0's and 1's, with at most one pair of consecutive 0's and at most one pair of consecutive 1's is shown in figure 2.23.

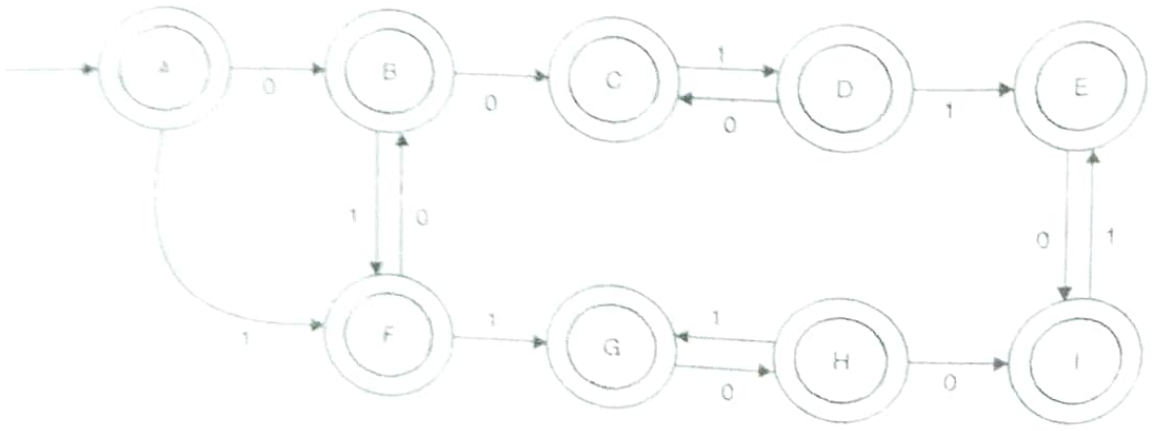

Fig. 2.23

2. Construct a F.A. which will accept string of 0's and 1's containing even number of zeroes and odd number of ones.

   A transition diagram of finite automata accepting the set of all strings of 0's and 1's, containing even number of 0's and odd number of 1's is shown in figure 2.24.

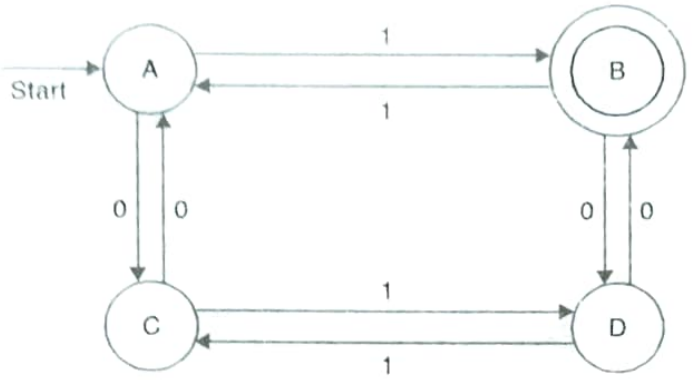

Fig. 2.24

3. Construct a F.A. which will accept string of 0's and 1's containing odd number of zeroes and even number of ones.

   A transition diagram of finite automata accepting the set of all strings of 0's and 1's, containing odd number of 0's and even number of 1's is shown in figure 2.25.

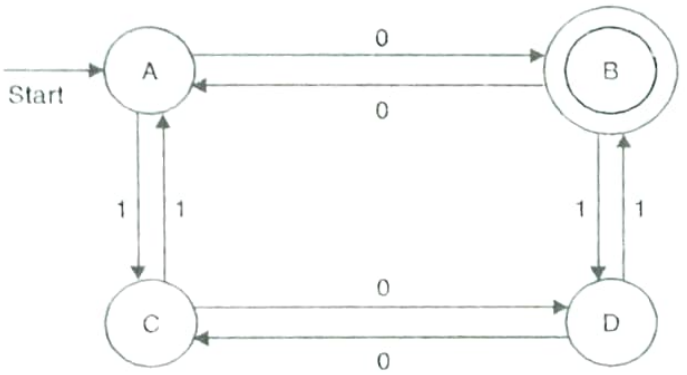

Fig. 2.25

4. Construct the F.A. for accepting strings of zeroes and ones containing equai number of zeroes and ones and no prefix of the string should contain two more zeroes than ones or two more ones than zeroes.

   A transition diagram of finite automata accepting the set of all strings of 0's and 1's, containing equal number of 0's and 1's and no prefix of the string containing two more zeroes than ones or two more ones than zeroes is shown in figure 2.26.

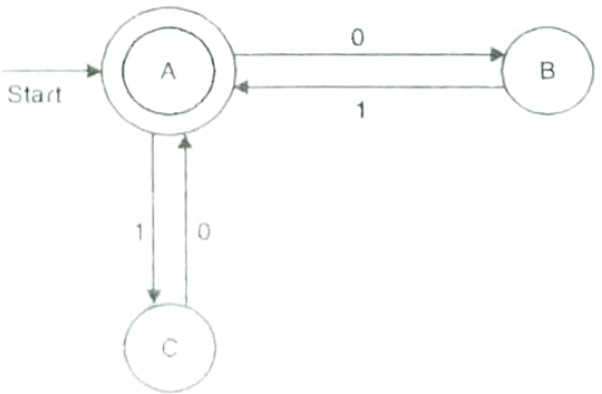

Fig. 2.26

5. Construct a F.A. for accepting all possible strings of zeroes and ones not contain 101 as a substring.

Figure 2.27 is a transition diagram of the automata accepting those strings that contain 101 as a substring.

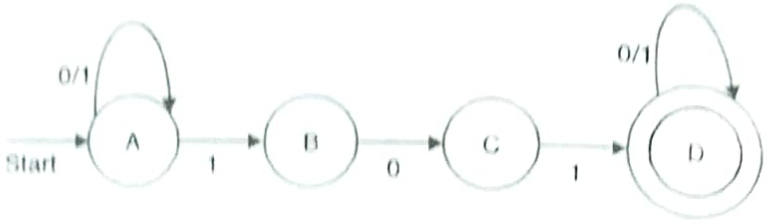

Fig. 2.27

A DFA equivalent to this NFA will be:

	0	1
{A}	{A}	{A, B}
{A, B}	{A, C}	{A, B}
{A, C}	{A}	{A, B, D}
{A, B, D}*	{A, C, D}	{A, B, D}
{A, C, D}*	{A, D}	{A, B, D}
{A, C, D}*	{A, D}	{A, B, D}

Let us identify the states of this DFA using the names given as follows:

{A}	q_0
{A, B}	q_1
{A, C}	q_2
{A, B, D}	q_3
{A, C, D}	q_4
{A, D}	q_5

The transition diagram of this automata is given in figure 2.28.

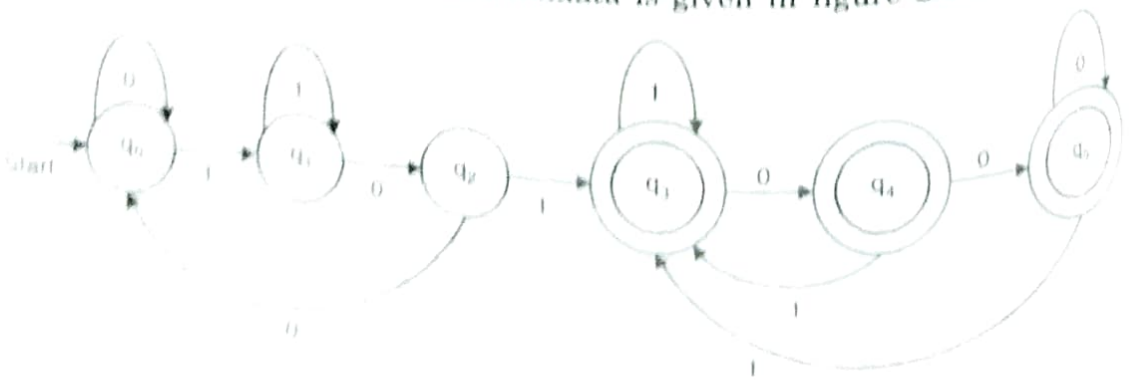

Fig. 2.28

5. Construct a F.A. for accepting all possible strings of zeroes and ones not containing 101 as a substring.

Figure 2.27 is a transition diagram of the automata accepting those strings that contain 101 as a substring.

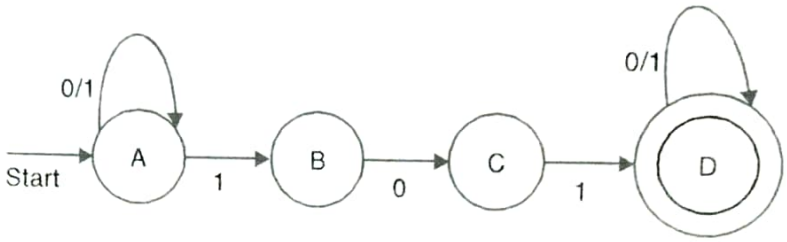

Fig. 2.27

A DFA equivalent to this NFA will be:

	0	1
{A}	{A}	{A, B}
{A, B}	{A, C}	{A, B}
{A, C}	{A}	{A, B, D}
{A, B, D}*	{A, C, D}	{A, B, D}
{A, C, D}*	{A, D}	{A, B, D}
{A, C, D}*	{A, D}	{A, B, D}

Let us identify the states of this DFA using the names given as follows:

{A}	q_0
{A, B}	q_1
{A, C}	q_2
{A, B, D}	q_3
{A, C, D}	q_4
{A, D}	q_5

The transition diagram of this automata is given in figure 2.28.

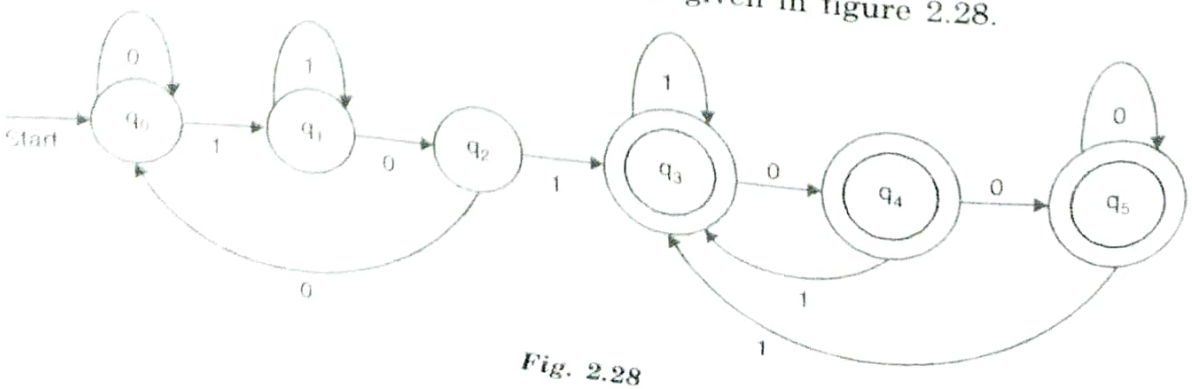

Fig. 2.28

The compliment of above automata is shown in figure 2.29.

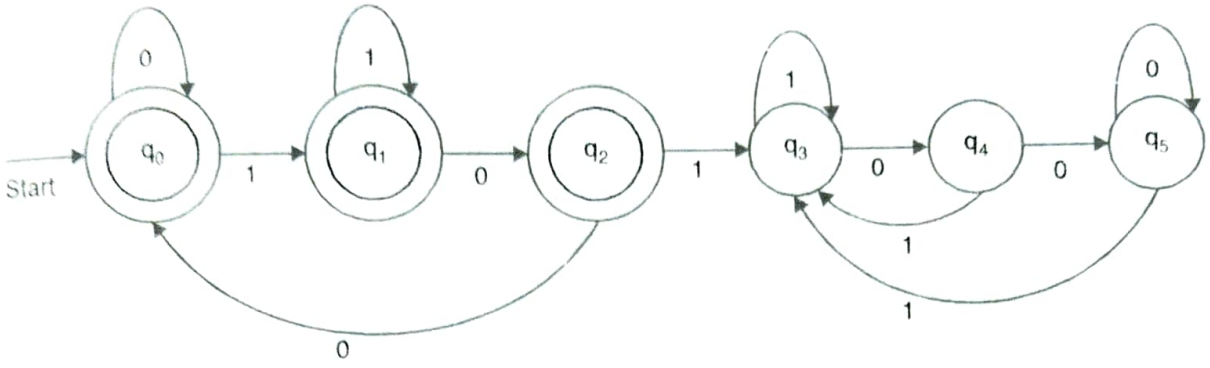

Fig. 2.29

After minimization we get the following DFA, because states q_3, q_4, q_5 are non-distinguishable states hence gets combined, and this combination becomes a dead state, hence can be eliminated.

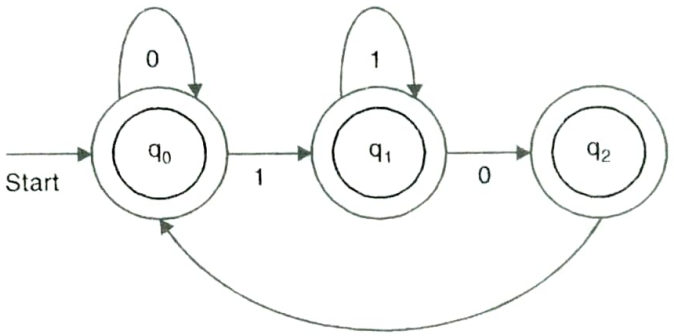

Fig. 2.30

6. Construct a F.A. which will accept those strings of decimal digits which are divisible by 3.

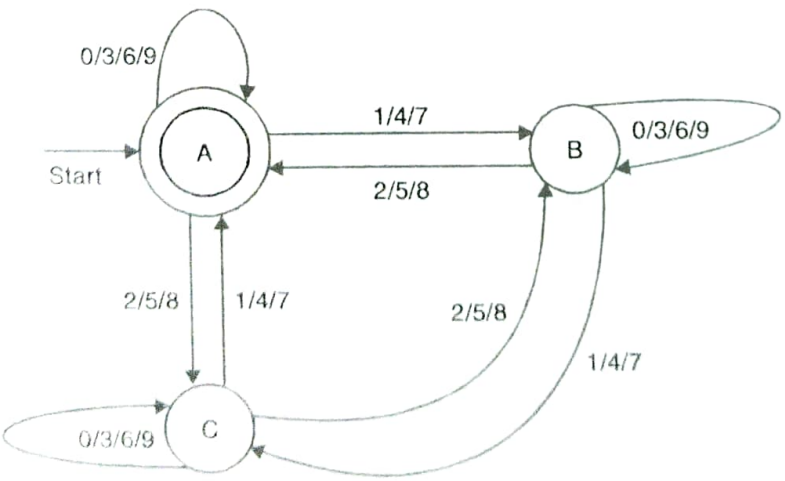

Fig. 2.31

7. Construct a F.A. for accepting all possible strings of zeroes and ones which does not contain 011 as a substring.

Figure 2.32 is a transition diagram of the automata accepting those strings that contain 011 as a substring.

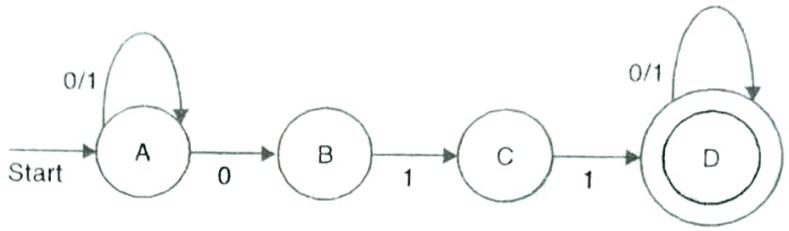

Fig. 2.32

A DFA equivalent to this NFA will be:

	0	1
{A}	{A, B}	{A}
{A, B}	{A, B}	{A, C}
{A, C}	{A, B}	{A, D}
{A, D}*	{A, B, D}	{A, D}
{A, B, D}*	{A, B, D}	{A, C, D}
{A, C, D}*	{A, B, D}	{A, D}

Let us identify the states of this DFA using the names given as follows:

{A}	q_0
{A, B}	q_1
{A, C}	q_2
{A, D}	q_3
{A, B, D}	q_4
{A, C, D}	q_5

The transition diagram of this automata is given in figure 2.33.

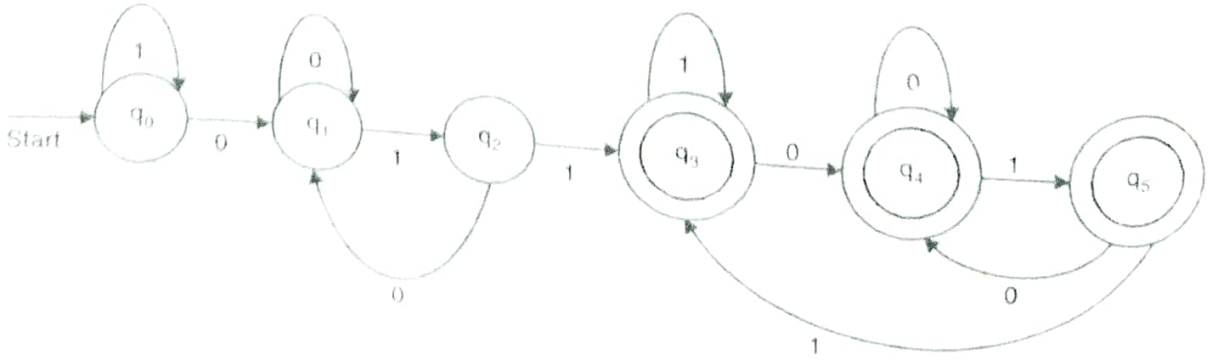

Fig. 2.33

The compliment of above automata is shown in figure 2.34.

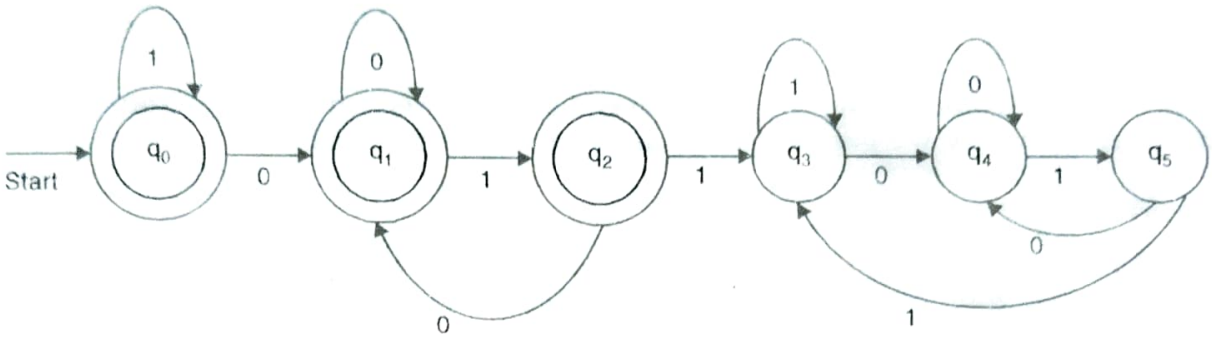

Fig. 2.34

After minimization we get the following DFA, because states q_3, q_4, q_5 are non-distinguishable states hence gets combined, and this combination becomes a dead state, hence can be eliminated.

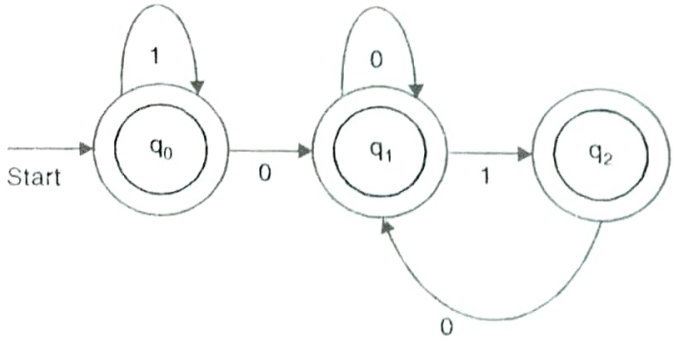

Fig. 2.35

8. Construct a F.A. which will accept those strings of binary number which are divisible by 3.

The transition diagram of this automata is given in figure 2.36.

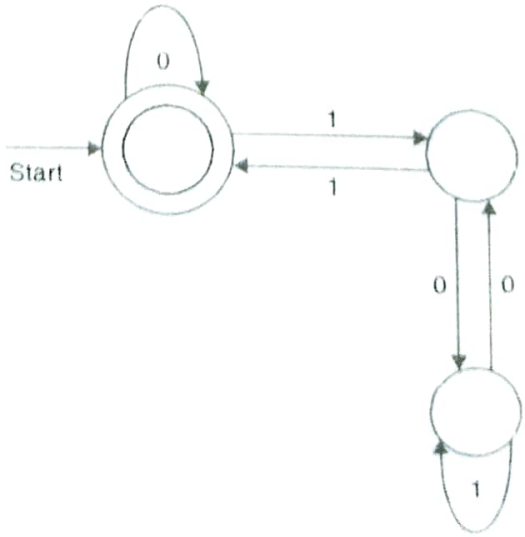

Fig. 2.36