Nondeterministic Finite Automata (NFA)[1]

1. A set of *states S*
2. A set of input symbols *S* (the input symbol alphabet)
3. A transition function, move, that maps state-symbol pairs to sets of states
4. A state, s_0, that is distinguished as the *start* or initial state.
5. A set of states, *F*, distinguished as accepting or *final* states.

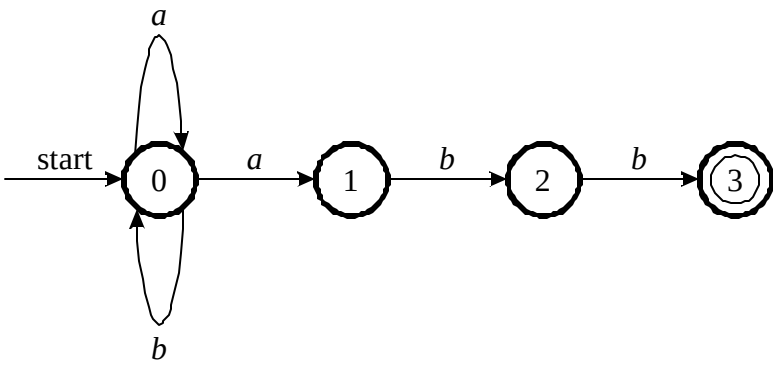

Figure 1. A transition diagram of a NFA that accepts (*a*|*b*)**abb*.

1. $S = \{0,1,2,3\}$
2. Table 1 shows the transition function move for the NFA of Figure 1.
3. $\Sigma = \{a,b\}$
4. $s_0 = 0$
5. $F = \{3\}$

State	Input Symbol	
	a	***b***
0	{0,1}	{0}
1	-	{2}
2	-	{3}

Table 1. Transition function for the NFA of Figure 1.

An NFA accepts an input string *x* if and only if there is some path in the transition graph from the start state to some accepting state, such that the edge labels along this path spell out *x*. For example, *aabb* is accepted by the path from state 0, following the edge labeled a to state 0 again, then to states 1, 2, and 3 via edges labeled *a*, *b*, and *b*, respectively.

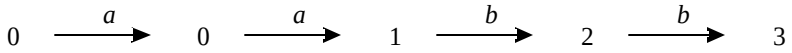

Figure 2. State transition diagram for the input string *aabb*.

[1] Excepted from Aho, Sethi, and Ullman *Compilers, principles, techniques, and tools*. Addison-Wesley, 1986, ISBN 0-201-10088-6

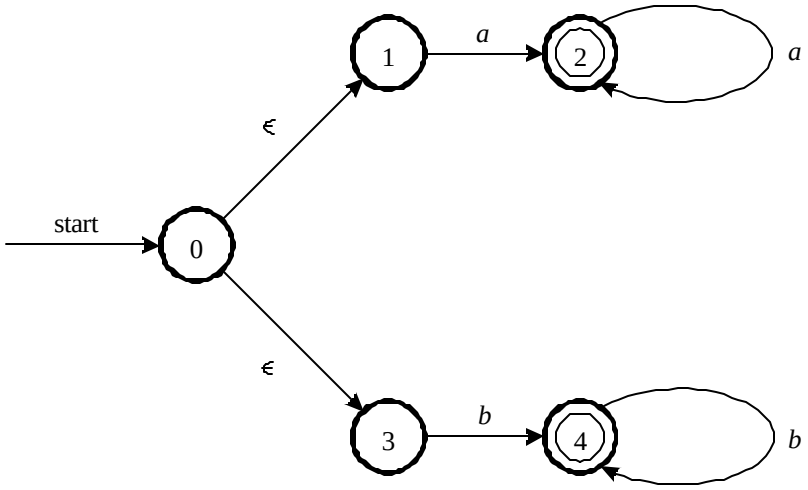

Figure 3. Nondeterministic Finite Automata that recognizes aa*|bb*

State	Input Symbol		
	Î	***a***	***b***
0	{1,3}		
1		{2}	
2		{2}	
3			{4}
4			{4}

Table 2. Transition function for the NFA of Figure 3.

1. $S = \{0,1,2,3,4\}$
2. Table 2 shows the transition function move for the NFA of Figure 3.
3. $\Sigma = \{\in, a, b\}$
4. $s_0 = 0$
5. $F = \{3,4\}$

Note that Î 's "disappear" in concatenation.

1. What is the input alphabet of each of the following languages?
   1.1. Pascal
   1.2. C
   1.3. Fortran 77
   1.4. Ada
   1.5. Lisp
2. What are the conventions regarding the use of blanks in each of the languages of exercise 1.
3. Describe the languages denoted by the following regular expressions:
   3.1. 0(0|1)*0
   3.2. ((Î |0)1*)*
   3.3. (0|1)*0(0|1)(0|1)
   3.4. 0*10*10*10*
4. Write regular definitions for the following languages.
   4.1. All strings of letters that contain the five vowels in order.
   4.2. Comments consisting of a string surrounded by /* and */ without an intervening */ unless it appears inside the quotes " and ".
5. Construct nondeterministic finite automata for the following regular expressions.
   5.1. (*a*|*b*)*
   5.2. (*a**|*b**)*
   5.3. ((Î |*a*)*b**)*
   5.4. (*a*|*b*)**abb*(*a*|*b*)*