1. Systems that implement regular expressions (such as flex) often have operators that make writing regular expressions easier. In the following are some examples of these operations. For each item write a regular expression that uses only the basic operations I showed in the lecture (concatenation RS, union R|S and closure R*) as translations of the operations shown below. In the following the alphabet is \{0,1,2,3,4,5,6,7,8,9\}

a. 1+ is a regular expression that specifies the language \{x \mid x \text{ is a sequence one or more } 1s\}

b. [2-5] is a regular expression that specifies the language \{2, 3, 4, 5\}.

c. [1-3]{2,} is a regular expression that specifies the language \{x \mid x \text{ is a string consisting of any combination of } 1, 2 \text{ or } 3s \text{ that is at least 2 characters long}\}. Some example strings are 11, 232, 3333, 12, 321, ...

d. [^1-5]+ is a regular expression that specifies the language \{x \mid x \text{ is a string containing characters } 6, 7, 8 \text{ or } 9 \text{ that is at least 1 character long}\}
2. Create an NFA for the regular expression \((1^*01^*0)^*\).