\* 1. Two languages *L*1 and *L*2 over the alphabet {0, 1} are given. Words of *L*1 are described by the expression 0\*1\*0\*1\*0\* and words of *L*2 are described by the expression (01+10)\*. Find

a) the shortest word in the intersection *L*1 ∩ *L*2, b) the longest word in the intersection *L*1 ∩ *L*2,

c) the shortest word which is in *L*1 and is not in *L*2, d) the shortest word which is in *L*2 and is not in *L*1,

e) the shortest word which is neither in *L*1 nor in *L*2.

\* 2. Draw a transition diagram of an automaton which accepts all words over the alphabet {0,1} which

a) contain the substring 1010 at least once b) do not contain the substring 1010

c) contain the substring 1010 exactly once d) contain the substring 1010 at most twice

\* 3. Write a regular expression describing a language over the alphabet {0, 1}

a) which words contain only 0's, b) each word of which contains exactly one 1,

c) each word of which contains at least one 1, d) each word of which contains at least two 1,

e) each word of which contains even number of 1's, f) each word of which contains odd number of 1's.

\* 4. Construct a NFA over alphabet {0, 1, 2} which detects in a text all substrings containing three 0's and two 1's.

\* 5. Construct a NFA over the alphabet {*a*, *b*, *c*, *d*} which detects in a text all occurences of the string in the format #*ba*##*b*#. The symbol # represents exactly one (and any one) symbol from the set {*a*, *b*, *d*}. The automaton must be able to process correctly a text of any length.

6. Construct an automaton over alphabet A = {a,b,c}, which will detect in a text all words characterized by the regular expression R = (ac\* + bb)\*a.

7. Consider the alphabet A= {*a*, *b*, *c*, ..., *z*}. Let us define the order of symbol *a* to be 1, the order of symbol *b* to be 2 etc, the order of symbol *z* to be 26. We say that a word over A is ordered if for any of its character *χ* holds that the order of every character which appears later in the word than *χ*  is higher than order of *χ*. Construct a NFA which detects in a text all occurences of all ordered words.

8. Construct a NFA over alphabet {0, 1, 2} which detects in a text all substrings in which the number of symbols 0, 1 and 2 is the same.

9. We would like to construct a NFA over the alphabet {0, 1} which accepts the language of all words which are certificates of some undirected tree. Decide whether such construction is possible and if it is describe it.

10. Consider the alphabet A = {*a*, *b*, *c*}. Operation ROT chooses an arbitrary character *x* in a given string and substitutes it by a character which immediately follows after x in A. When x is the last character in A the ROT operation substitutes it by the first character in A. Construct a NFA which detects in a text all such substrings which can be obtained from the pattern *aabcb* by applying operation ROT at most twice.

11. Decide whether two given regular expression represent the same regular language.

a) (01+0)\*0 b) 0(10+0)\*



12. Characterize informally the language over the alphabet {0, 1} accepted by the given automaton. Write a regular expression describing this language.

13. Write a regular expression which describes maximum set M of the strings over alphabet {*a*, *b*, *c*}. It holds for M:

a) Each string in M begins and starts with symbol *b*.

b) Each string in M contains exactly one symbol *c* .

c) No string in M contains symbol *a* at odd position. Positions are indexed from 1.