

Automaton minimization I.

| | | |
|----------------------|-------------------------------|-----------------|
| Submission deadline: | 2011-11-19 23:59:59 | 2764723.489 sec |
| Evaluation: | 0.0000 | |
| Max. assessment: | 3.0000 (Without bonus points) | |
| Submissions: | 0 / 20 | |
| Advices: | 0 / 0 | |

First task

Your task is to determinize and minimize a finite automaton.

1. Determinize the given nondeterministic finite automaton (description of its format follows).
2. Minimize the resulting deterministic automaton.

Submit the minimized automaton in the form of a text file in the format described below.

Your automaton must meet the following criterion: either the delta transition function of the minimized automaton is total (that is, defined for all pairs <state,input symbol>), or the minimized automaton does not contain any redundant state (that is, state from which you can never get to a final state). This criterion is necessary for an algorithm that checks validity of your minimization. For simplification of notation of the automaton, you can rename its states.

Text format of the automaton:

The file with the automaton contains its transition table (multiple rows, each consisting of multiple cells). Individual cells in a row are separated by at least one space. Input symbols are denoted by a single character. Only exception is \eps for epsilon. State can be labeled by a string of characters consisting of alphanumeric (ASCII 7-bit) characters.

The first line begins with an identifier DFA or NFA for deterministic or nondeterministic automaton. Further cells contain the input symbols - the first line describes a header of the transition table. The second row and the following rows begin with a name of state, optionally preceded by a marker of start (>) and/or final (<) state. There is exactly one start state and there can be multiple final states.

The first column thus contains all source states of the transitions. The source state in every row is followed by lists of target states for every input symbol. The list can contain no target state (-), one target state, or multiple target states - the states are then separated by the pipe character (|). Multiple target states for one input symbol are not separated by spaces!.

Examples:

Nondeterministic:

```
NFA  a  b  c  \eps
><A  B  -  C | A  -
      B  A  A  A | B  A
<C   C  C  C   -
```

And deterministic one:

```
DFA  a  b  c
><A  B  D  C
<B   B  A  C
<C   C  C  C
      D  D  D  D
```

Since you are asked to determinize your automaton, your solution file shall begin with abbreviation DFA.

Submit your file in plain ASCII encoding and do not use any special characters for state names.

Your NFA to process:

```
NFA a b c d \eps
>0 26 1|6|21 11|16 - -
  1 - - 2 - -
  2 3 - - - -
  3 - - - 4 5
  4 - - - 5 0
<5 - - - - -
  6 - - 7 - -
  7 - - - 8 -
  8 - - 9 - -
  9 - - - 10 -
<10 - - - - -
  11 - - - 12 0
  12 - - - 13 14
  13 - - - 14 -
  14 - - - 15 -
<15 - - - - -
  16 - - - 17 -
  17 - - - 18 -
  18 - - - 19 -
  19 - - - 20 -
<20 - - - - -
  21 - - 22 - -
  22 - - - 23 24
  23 - - 24 - -
  24 - - - 25 -
<25 - - - - 22
  26 - 27 - - -
  27 - 28 - - -
  28 - - 29 - -
  29 - - - 30 -
<30 - - - - -
```

Submit:

Submit