

# SUPERUTILS.COM UBER TURING MACHINE V1.3 (SEPTEMBER 13, 2011) HELP DOCUMENTATION

This file is a complete user manual for the Uber Turing Machine software. You may wish to print the contents of this document for your reference at any time.

## Contents

Fundamentals.....	1
Allowed Symbols.....	2
Definition of the Algorithm.....	2
Adding a Row.....	3
Deleting a Row.....	3
Adding a Column.....	3
Deleting a Column.....	4
Changing of the Blank Symbol.....	4
Inputting Instructions.....	4
Saving/Loading of an Algorithm.....	5
Definition of the Tape.....	5
Adding Cells.....	5
Deleting a Cell.....	6
Setting-up Head Position.....	6
Saving/Loading of a Tape.....	6
Definition of the Execution.....	6
Verifying a Program.....	6
Normal Mode.....	6
Single-step Mode.....	7
Quick Mode.....	7

## Fundamentals

Creating and editing a Turing machine program involves dealing with its parts: the algorithm with alphabet and data on the tape. Adding new tape cells, also columns, and states (rows) is performed in a dialog mode. Editing of contents is done directly from the main window of Uber Turing Machine. There are also features of saving and loading programs on/from a disk.

The program may be executed only if it's syntactically and semantically correct. Program's verification is done every time before the execution.

There are three execution modes in the simulator: normal, single-step, and a quick mode. The modes differ in execution speed (by number of performed instructions per time unit), and in amount of output statistics. Thus, the quick mode is intended for fast computing of program's results, or for determining possible infinite loops in the algorithm. The single-step mode is designed for studying and debugging Turing machine programs. The normal mode provides a detailed log on every program step, and enables you to regulate delay between two steps.

## Allowed Symbols

The only following ASCII characters are allowed to be used as symbols within the simulator:

Character	Decimal Code	Hexadecimal Code
!	33	21h
#	35	23h
\$	36	24h
%	37	25h
*	42	2Ah
+	43	2Bh
0–9	48–57	30h–39h
=	61	3Dh
?	63	3Fh
@	64	40h
A–Z	65–90	41h–5Ah
^	94	5Eh
_	95	5Fh
a–z	97–122	61h–7Ah

## Definition of the Algorithm

The Turing machine algorithm, in this particular implementation of the simulator, is a set of transition rules, recorded and edited in a tabular form. Table rows define quantity of used states, and columns define processed tape symbols. At the crossing of i-th row and j-th column, there is an instruction that tells, what to write in a cell of the tape, where to move over the tape (to the left, to the right, or stay at the same position), and what state number should be on the next algorithm step. This instruction is being called only if the machine currently in a state number i, and a current read symbol on the tape is j.

When creating an algorithm from scratch, you can change a number of states and symbols of the

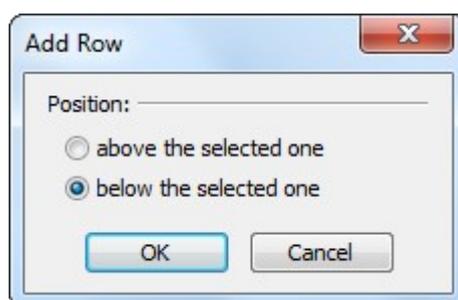
alphabet, by adding rows and cols at any place of the table. Note, that when you add a row to the middle of the action table, it causes automatically renumbering of all existing instructions to keep logic of the algorithm.

When inputting instructions, please keep the following format: <New symbol> <Shift> <Next state>. The arguments must be separated by a whitespace character.

The alphabet of the Turing machine is formed from processed symbols automatically, but you must specify a special blank symbol. It is intended, that the tape is indefinitely extensible to the left and to the right, and is filled by this blank symbol.

## Adding a Row

When adding a new row (state) to the action table, at first click above or below of the desired position. Then call the menu command **Algorithm** → **Add Row**, or click the button  on the toolbar, or press <Insert>. The Add Row dialog window will appear on the screen:



Choose a position of the inserting row and click OK.

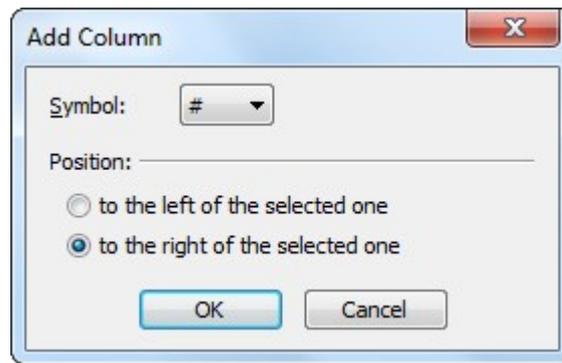
## Deleting a Row

For deleting a row, at first select it by clicking on one of its cells, then call the menu command **Algorithm** → **Delete Row**, or click the button  on the toolbar, or press <Ctrl>+<Delete>. If the deleting row was the only one in the algorithm, the warning message will appear.

When deleting, the automatic correction happens among already defined instructions: rules, that had transitions to the deleted state, are cleared, and all rules after deleted are renumbered.

## Adding a Column

For adding a new column — a new processing symbol — at first define its position in the table by clicking on a table cell to the left or to the right of it. Then call the menu command **Algorithm** → **Add Column**, or click the button  on the toolbar, or press <Shift>+<Insert>. The following dialog window will appear on the screen:



In the window choose one of the allowed and yet-not-used symbols, define a position of the inserting column, and then click OK.

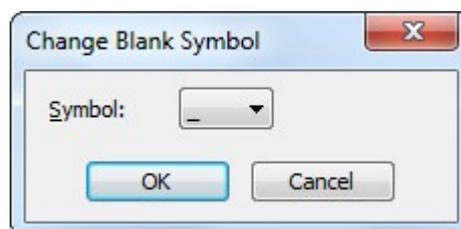
### Deleting a Column

For deleting a column, at first select it by clicking on one of its cells. Then call the menu command **Algorithm** → **Delete Column**, or click the button  on the toolbar, or press <Shift>+<Ctrl>+<Delete>. If the deleting column defined rules for the blank character, the warning message will appear.

Deleting of a column provokes automatic correction of the action table: all instructions, that write the deleting symbol, will be cleared.

### Changing of the Blank Symbol

The blank symbol plays an important role in the Turing machine abstraction — it defines a symbol which fills all the indefinite tape to the left and to the right. For changing the blank symbol, choose the menu command **Algorithm** → **Change Blank Symbol**, or click the button  on the toolbar, or press <Ctrl>+<B>. The following dialog window will appear on the screen:



In the window choose a new blank symbol among all used in the algorithm, then click OK to apply changes.

### Inputting Instructions

Inputting rules is performed by typing-in instructions right into the action table. But please keep the following format: <New symbol> <Shift> <Next state>, where:

- New symbol — is a symbol to write on the tape.
- Shift — a direction to move the head over the tape. Coded by one of the following symbols: L, R, N (shift to the left, right, no shift).
- Next state — a state number for the next state (0 is the final state, it halts program's execution).

The arguments must be separated by a whitespace character (the 32nd code point in the ASCII table).

## Saving/Loading of an Algorithm

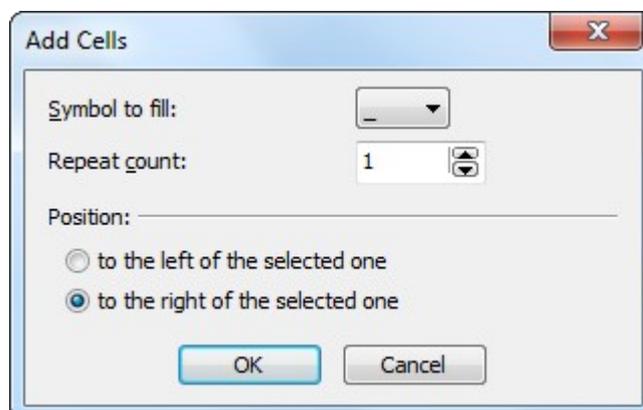
For your convenience, there are features for saving and loading a Turing machine algorithm on/from a disk. Check it out: in the menu **Algorithm**, there are items **Load Algorithm** and **Save Algorithm**. The **Save Algorithm As** menu command is intended for saving the algorithm under a new filename.

## Definition of the Tape

In this particular implementation of the Turing machine, the tape is indefinite on both sides, and is limited only by an available physical memory of a computer. In the simulator you are able to define initial data on the tape, and a start position of the head.

## Adding Cells

For adding new cells onto the tape, at first define inserting position by clicking to the left, or to the right of the desired spot. Then call the menu command **Tape** → **Add Cells**, or click the button  on the toolbar, or press <Shift>+<Ctrl>+<Insert>. The following dialog window will appear on the screen:



Choose a symbol to fill from the drop-down list and define its repeat count. Then select the

inserting position and click OK to apply.

## Deleting a Cell

For deleting a selected cell, call the menu command **Tape** → **Delete Cell**, or click the button  on the toolbar, or press <Ctrl>+<D>. If the deleting cell on the tape was the last one, the warning message will appear.

## Setting-up Head Position

For defining a start position of the head, simply select a needed cell of the tape. When being executed, the algorithm will start working from that cell.

## Saving/Loading of a Tape

There are features for saving and loading tape data into/from a file. They are available in the menu **Tape**, under the names **Save Tape** and **Load Tape**. The **Save Tape As** menu command is intended for saving the tape data under a name filename.

## Definition of the Execution

Program's execution is done step-by-step, according to the programmed algorithm, thereby changing data on the tape. There are three execution modes in the simulator: normal, single-step and a quick mode. Every time before doing the execution, verification of the program is performed.

## Verifying a Program

Verifying of a program is needed to define its correctness and to check cells on the tape against the used alphabet. After that, the algorithm is also checked for at least one transition into the final state.

To manually call the verification, click the button  on the toolbar, or press <Ctrl>+<F9>. If there will be one of the warning messages, you may not execute the verified program.

## Normal Mode

To start execution of a program in a normal mode, call the menu item **Run** → **Start**, or click the button  on the toolbar, or press <F9>. After verification, if the program is correct, the step-by-step simulation process will be started. When executing, the status of the machine is displayed in the action table and on the tape, by showing a currently executed instruction and a current symbol on the tape. You may change speed of the process by dragging the speed slider on the main toolbar.

To pause the simulation, click the button  on the toolbar, or call the menu command **Run** → **Pause**, or press <Ctrl>+<P>. By doing that, you will be turned out in a single-step mode. In the single-step mode you may execute a current instruction by pressing <F8> (or by clicking the button  on the main toolbar, or by calling **Run** → **Make Step**). In every moment you can return in the normal mode by calling the **Start** command.

The simulation will be ended automatically when the machine will come into the final state. When ending, complete output statistics will be printed out into the Reports window. If the program in not ended after a long period of time, it may be considered as infinitely looped, so you may break it by calling the **Run** → **Stop** command, or by clicking the button  on the main toolbar, or by pressing <Ctrl>+<F2>.

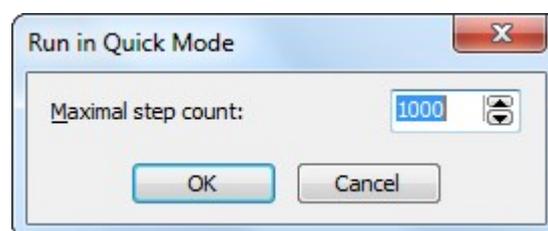
## Single-step Mode

You can start executing a program in a single-step mode by calling the menu command **Run** → **Single-step Mode**, or by clicking the button  on the toolbar, or by pressing <Shift>+<Ctrl>+<F9>. It is an equivalent of starting a program in the normal mode, and then immediately pausing it.

## Quick Mode

This mode is intended for quick carrying out program results, and for detecting possible indefinite loops in the algorithm.

For running a program in the quick mode, click the button  on the toolbar, or call the menu command **Run** → **Quick Mode**, or press <Ctrl>+<Q>. If the program is correct, the following dialog window will appear on the screen:



Type-in maximal algorithm step count into the field and click OK. If the program will not be finished after the specified number of steps, the simulation will show a request dialog for continuing of execution.