The Turing-in-a-Flash Light

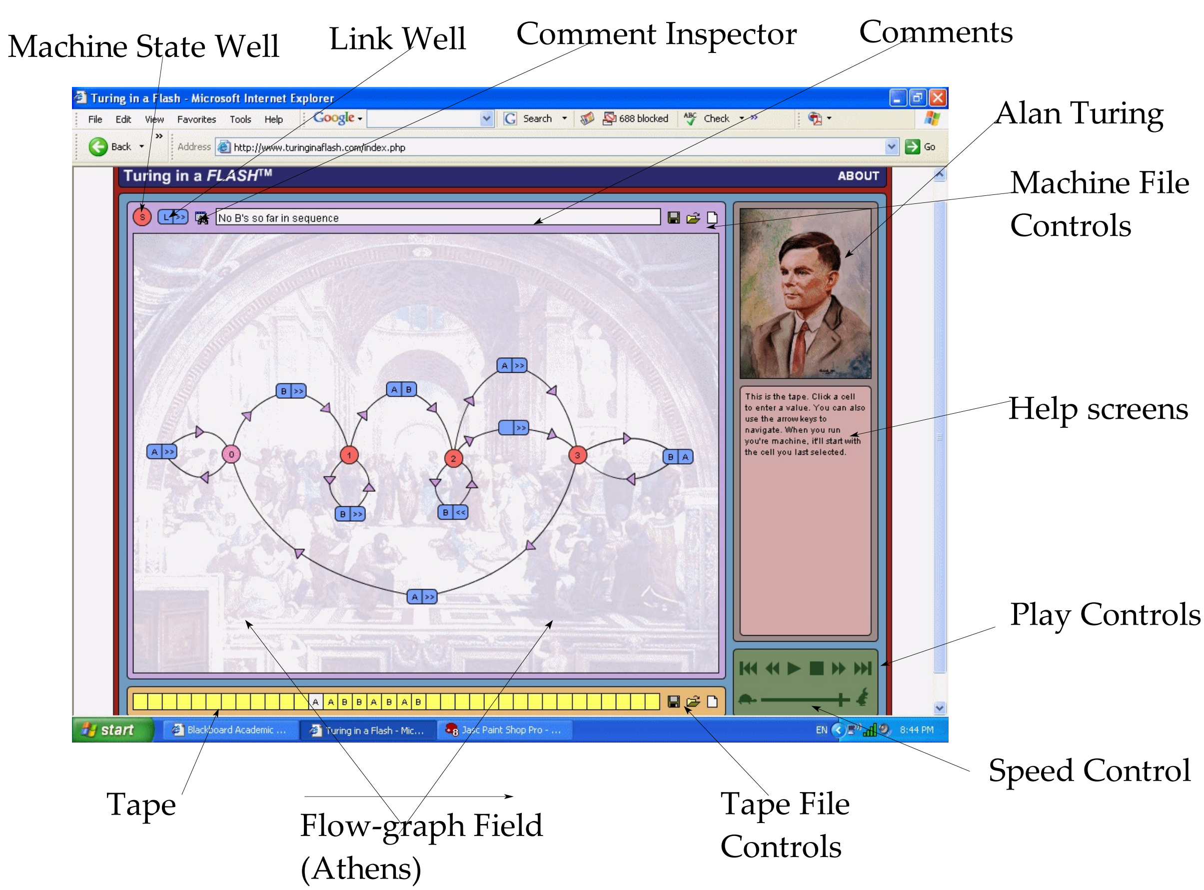
Turing-in-a-Flash is a web-based Turing machine emulation. It allows the user to design Turing machines using flow-graphs and then watch them execute. Both machine designs and tapes can be downloaded from the web or uploaded to the machine for execution. The speed with which the machine operates is variable. Turing-in-a-Flash provides ample facility to understand the nature and function of Turing machines and practice their design, which are great ways to approach general issue in computation.

How to Get into the Program

The link to the program site ([http://www.turinginaflash.com)](http://www.turinginaflash.com) is in the Blackboard site under Assignments. Just click on it, and it will come up in a new window, ready to go.

Understanding the Screen

Once the program is up on the computer, a screen that looks like this will be displayed in your web browser.



How to Load and Save Machines

On your own computer, create a new folder (call it "Turing in a Flash" or "TiaF" or "447," whatever) in which to save machine designs and test tapes. For assignment 1, go to Bb, Assignments, *right* click on "Lone Rearranger," then click on "save target as" and save it to the folder just created. To upload it then into TiaF, click on the "open file" icon, browse to find "Lone Rearranger" in the TiaF folder, select it, and click "open." You should then have the TiaF screen displayed, with the machine design displayed in the flow-graph field.

To save a machine you have created, click on the "save file" icon. You will get a "File Download" window asking if you want to save the file. Click on "Save," and a "Save as" window will open. Check to make sure that you're going to save your machine to the right folder, then put in the name you want for the saved file. For the machine designs that you'll be sending me, make sure that the name you give the file contains your name and the lab number.

Once you have saved a machine to your computer, you can, of course, load it back into TiaF. So, if you start work on a machine and get interrupted, you can save it and pick it back up when you can.

The tape file control that looks like a "new document" icon will erase everything in the flow-graph field. This is the nuclear option, if one feels the need to begin all over from the beginning.

How to Save and Load a Tape

Saving and loading tapes is just like saving and loading machines, except that one uses the special buttons for tapes. Store them in the same folder created for your machines, or create a subfolder just for tapes.

How to Run a Machine

These controls operate much like the tape/CD controls they look like, with one exception. The "play" button starts the machine going, and the "stop" button stops it (well, duh!). Here's the exception: The double triangle (fast forward or rewind) does not actually fast forward—it allows one to execute a single operation, one step forward or one step back. The "first track/last track" buttons takes one immediately to the beginning of the whole thing (tape and execution) or all the way to the completed execution of the machine. The speed with which the machine runs can be changed by sliding the speed control.

To start the machine, (1) click on the square on the tape where the machine should start, (2) make sure the state node is selected at which the machine should start (usually state 0, but when testing a machine under design, it may be convenient to start a test run with a later state), then (3) hit the play button. The green rectangle behind the play buttons will lighten: as long as that rectangle is the lighter shade, the program is expecting play/stop commands, and other functions (adding or changing states or links, changing the tape) will be disabled. Click on the stop button; the green rectangle will darken, and states, links, and tape symbols can be added or changed.

How to Create a Tape

Click on the square that is intended to be the leftmost square of the symbol string to be placed on the tape, then simply type in the symbol string. The machine recognizes pretty much all the characters available on a standard keyboard. The backspace and delete keys work as one would expect. The entire tape can be erased with one click on the "New Tape" button—the blank sheet.

How to Create a Machine

Start by going to the *Machine State Well*, clicking on it and dragging a machine state onto the flow-graph field. The first state dropped will automatically be numbered state 0. Click and drag as many states as necessary or convenient to begin. They will automatically be numbered in sequence. Once a Machine State node is on the flow-graph field, it can be repositioned at any time by simply clicking on it and dragging it to its new location.

To create the links between Machine State nodes, click on the *Link Well*and drag a link onto the field, between the two nodes to be connected (or, if the link is to be a loop on a single node, place it near that node). Little arcs with blue circles at the ends will pop out of the link capsule: drag the blue circles to the appropriate nodes to be linked and drop them. The links originally appear with a default direction or flow from left to right. Be careful that arrows on the links point from the initial machine state node to the target machine state node. This may require dragging the arc initially on the left around to the right side, and the arc initially on the right around to the left.

Once the link has been created, click on the link capsule again and type the appropriate symbols in. The left side of the capsule will contain the symbol that must be on the tape if that link is to be activated in the initial machine state. The right side of the capsule will contain the designation of the action to be performed when in that initial machine state, reading the symbol designated in the left side of the capsule. If the action to be performed is printing a symbol on the tape, simply type the relevant symbol in; if the action is moving left or right, hit the relevant arrow key on your computer.

Links can be repositioned in numerous ways. The link capsule can be moved by simply clicking and dragging, and the right and left arcs can be repositioned by clicking and dragging on the direction-arrows in them. It should not be difficult to make the machine relatively easy to "read." If testing the machine leads to a design change, a link can easily be moved to anchor in a different machine state: click on the relevant link capsule to select that link, then simply click and drag the blue circle (coincident with the old node connected to that link) over to its new home machine state.

To clear an unwanted link, click on the capsule to select it, then hit either "delete" or "backspace." To clear an unwanted machine state node, select it and hit either "delete" or "backspace."

Commenting on a Machine

Like any good software design, machines need commentary. One can sometimes tell what a simple machine is supposed to do by looking at it, but complex machines are very rarely obvious, either to design or to diagnose. The point of comments is to indicate the design intentions for the different pieces of the machine. In TiaF, one can comment on the machine states and the links. Do NOT put into the comments the information that can simply be read off the machine diagram itself (e.g., in machine state 0, it will move right as long as it sees A's). Comments should tell the reader what the general point of that machine state or link is. Normally, commenting on states alone is adequate; links need to be commented only when particularly interesting or important.

The text box directly underneath the tape is where one puts one's comments. They will automatically be associated with the currently selected machine state node or link. Keep comments short and pithy. The full set of comments for a machine can be viewed by clicking on the "Comment Inspector."

Some Strategies for Designing Turing Machines

I find that the biggest single problem people have in designing these machines is forgetting how incredibly stupid they (the machines, that is) are. The only things these machines can do are read one symbol at a time and either print something in that square or move left or right one square. So complex tasks must be broken down into sequences of very simple, very moronic steps. Furthermore, at any moment, what the machine does is dependent *entirely* on two and only two things: (1) what machine state it is in and (2) what symbol it reads on the tape. The machine cannot by itself remember how many p's or q's it has seen; it cannot by itself find any particular square on the tape, unless there is something quite distinctive about that square. Even mind-bogglingly simple tasks, such as finding the second 'A' in a string of letters requires a string of instructions.

Next, remember that what is important about a Turing machine is almost always what's on the tape. As long as it produces the right output on the tape for the given input, exactly how it does it is not crucial: I'm not grading on the elegance of the machine, just its functionality. Do not get too caught up in the flow diagram itself: it is merely a means to an end. Focus, therefore, on what is happening on the tape, and what you could do next to take another step towards transforming the tape into what it is supposed to end up being. As you design your machine, test it frequently as it grows, making sure that at each step it is doing what you want it to do. In larger machine, test runs need not start off from machine state 0 every time. Simply select a higher-number state and the appropriate tape square to begin the run from.

Repetitive tasks usually take loops with some stop condition. For instance, getting to the end of a string of symbols takes a loop (or if there is more than one character in the string, a set of loops) that keep the machine moving in the right direction, with the stop condition being the blank square that bounds the string. Complex repetitive tasks require complex loops.