Turing Machine Development Environment

Daniel Guetta
Mentor: Bobi Gilburd
Faculty of Computer Science
The Turing Machine is a simple computing model equivalent to today’s computers.

It can show us what computers can do and what they can’t do – in other words, whether a task is computable or not.
Computability

\[ f(x) = x + 1 \text{ is computable} \]

Deciding whether a program will ever halt is not computable
What is a Turing Machine?

Read/write head

A B A 1 0 # G D N A B 0 1 1 ...

Semi finite tape
How do you tell a Turing Machine what to do?

Each Turing Machine has a set of states.

If Machine is in state \( x \) and is reading character \( a \), change to state \( y \), change the character to \( b \) and move left, right or not at all.
A simple example – addition

<table>
<thead>
<tr>
<th>State (q)</th>
<th>Transition</th>
<th>State (q)</th>
<th>Transition</th>
<th>State (q)</th>
</tr>
</thead>
<tbody>
<tr>
<td>q₁</td>
<td>q₁, 1, R</td>
<td>q₁, 1, R</td>
<td>q₂, _, L</td>
<td></td>
</tr>
<tr>
<td>q₂</td>
<td>q₃, _, L</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>q₃</td>
<td>FINAL STATE</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5 + 3 = 8
The basic Turing Machine model presented so far can have several variations.

A particularly interesting variation of the Turing Machine is one that has multiple tapes.
Multitape Turing Machines

1 1 1 1 1 + 1 1 1

1 1 1 1 1 + 1 1 1

= 

1 1 1 1 1 + 1 1 1
Equivalence of the Multitape Model
Aims of program – improvement over existing work

- To perform the following conversions:

<table>
<thead>
<tr>
<th>STATE</th>
<th>TRUTH TABLES</th>
<th>PSEUDOCODE</th>
<th>INSTRUCTIONS</th>
<th>STATE DIAGRAMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>q₁</td>
<td>1, r, 1, R</td>
<td>q₁, 1, R</td>
<td>If in q₁ reading 1, write 1, move right goto q₁.</td>
<td>MULTITAPE MACHINES</td>
</tr>
<tr>
<td>q₂</td>
<td>_, L</td>
<td>_, L</td>
<td>If in q₂ reading 1, write _, move left, goto q₃.</td>
<td>MULTITAPE SIMULATED</td>
</tr>
<tr>
<td>q₃</td>
<td>FINAL STATE</td>
<td></td>
<td></td>
<td>ON SINGLE TAPE</td>
</tr>
</tbody>
</table>
Aims of program – improvement over existing work – continued

- To simulate Turing Machines in a graphical environment
- The ability to save machines and re-open them later
- To divide the program into parts that can then be used independently in other programs.
- The ability to view, edit and/or simulate several machines at the same time, for comparison.
Demonstration of the program

You will now see a short video of a user using the program to create machines
Future work

- Implement the theory set out for the pseudo code → Turing Machine conversion
- Support different Turing Machine variations (doubly infinite, multi-track, etc…)
- Include finite state automatons in the program
Acknowledgements

- My mentor, Bobi Gilburd
- Technion’s computer science faculty
- My father and Stanislav Tsanev
- The British Technion Society
- Everyone that made SciTech 2003 such an amazing, incredible experience