**LMC Assembly**

Because machine code is hard for humans to read and write well, assembly code was created. Assembly is human readable machine code. Each assembly instruction directly corresponds to a machine instruction, but we use words instead of numbers to represent them. Also, instead of having to worry about exact memory locations where we are going to store data, we are able to give memory locations names. Thus we can say things like "Store this in the place I called ANSWER" instead of "Store this in location 9". This makes it much easier to keep track of what operations are happening to what memory and much easier to modify programs (don't have to worry about relocating our data memory if our program gets bigger.

Here is a LMC program in machine codes and assembly:

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Instruction** | **Assembly** | **Machine** |
| 0 | Input | INP | 901 |
| 1 | Store accumulator in location called FIRST | STA FIRST | 306 |
| 2 | Get input | INP | 901 |
| 3 | Add value from location called FIRST to the accumulator (which currently has second number) | ADD FIRST | 106 |
| 4 | Output the value in the accumulator | OUT | 902 |
| 5 | Stop | HLT | 0 |
| 6 | This is the location called FIRST. It stores DAT (Data). The initial value is 0. (If you do not write an initial value, 0 is assumed). Automatically set to location 6 because that is where it falls. If we added instructions above it, it would be a new location. (But we would not care since we don't ever call the location by number). | FIRST DAT 0 | 0 |

Assembly is still pretty hard to read and write (especially to do complex tasks), but it is much more friendly than machine code.

To try writing assembly, fire up this LMC simulator:  
<http://www.yorku.ca/sychen/research/LMC/LittleMan.html>

Then copy and paste this code:  
INP

STA FIRST

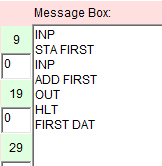
INP

ADD FIRST

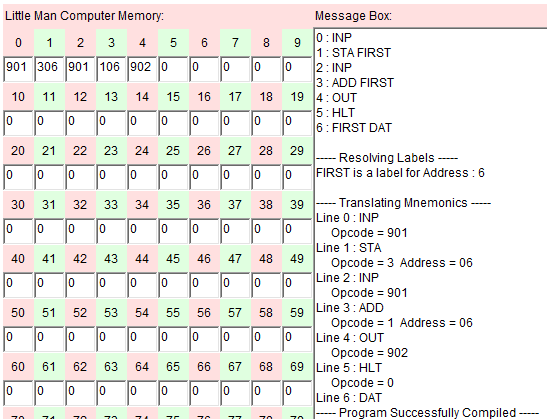
OUT

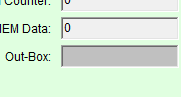
HLT

FIRST DAT



Into the Message Box:

Then press the Compile Program button. The Message Box will show the progress as it converts your assembly to machine code. The program you wrote will be placed into memory. 

You can use this simulator to run your code (it should add two input values). Your answer will appear in the Out-Box area of the window.

Or you can take the numeric machine code and use the excel LMC simulator to run it.

You can find a reference for LMC Assembly here:

<http://www.yorku.ca/sychen/research/LMC/LMCInstructions.html>

You can find more sample programs here:

<http://www.yorku.ca/sychen/research/LMC/index.html>

Click on one of the examples, then find the assembly code and copy and paste it into the MessageBox. Make sure to clear the message box before pasting in your code (you can't have the output information from your last program in the box, it should only have your new code).