Turing Complete – Calculations (Kin’s buffed version) in Python

Documentation

1. Registers
   1. There are 16 registers, marked from 0000 (1) to 1111 (15) in 4-bit binary.
   2. All registers start with 0.
   3. All registers’ values range from 0 to 255.
2. Syntax (in binary)

If you aren’t familiar with binary, check part 3.

|  |  |
| --- | --- |
| Meaning | Binary syntax & Full meaning |
| Immediate | 0000000000000000 |
| Copy from input (internal) | 0010[reg][inp]  Copy from [inp] to [reg], [inp] is 8 bits binary number, [reg] is 4 bits binary number. |
| Copy to output | 0100[reg]00000000  Copy from [reg] to the output stream. |
| Copy from input (external) | 0110[reg]00000000  Copy from input stream to [reg]. |
| Copy from [reg] to [reg] | 1000[reg][dist]0000  Copy from [reg] to [dist], [dist] is the destination register. |
| Do math | 1010[ins]000000000  Do math from reg. 0000 and reg. 0001 to reg. 0010, [ins] is 1 of the following:   |  |  | | --- | --- | | Binary code | Meaning | | 000 | Add | | 001 | Subtract | | 010 | Multiply | | 011 | Divide | | 100 | Exponent | | 101 | Root (reg. 0000 root reg. 0001) | |
| Do condition | 1100[ins]000000000[command]  If reg. 0000 [ins] reg.0001 then execute [command], [command] is in part 2 (except this), [ins] is 1 of the following:   |  |  | | --- | --- | | Binary code | Meaning | | 000 | Greater than | | 001 | Smaller than | | 010 | Not greater than | | 011 | Not smaller than | | 100 | Equal to | | 101 | Not equal to | |
| Do bitwise things (logic) | 1110[ins]000000000  Do logic from reg. 0000 and reg. 0001 to reg. 0010 (In the NOT case, reg. 0000 is the only input, output is the same), [ins] is 1 of the following:   |  |  | | --- | --- | | Binary code | Meaning | | 000 | AND | | 001 | NOT | | 010 | NAND | | 011 | OR | | 100 | NOR | | 101 | XOR | | 110 | XNOR | |

1. Syntax (in assembly-like language)

|  |  |
| --- | --- |
| Meaning | Assembly-like syntax & Full meaning |
| Immediate | imm |
| Copy from input (internal) | cin [reg] [inp]  Copy from [inp] to [reg], [inp] is from 0 to 255, [reg] is from 0 to 15. |
| Copy to output | out [reg]  Copy from [reg] to the output stream. |
| Copy from input (external) | cun [reg]  Copy from input stream to [reg]. |
| Copy from [reg] to [reg] | cpy [reg] [dist]  Copy from [reg] to [dist], [dist] is the destination register. |
| Do math | mth [ins]  Do math from reg. 0 and reg. 1 to reg. 2, [ins] is 1 of the following:   |  |  | | --- | --- | | Binary code | Meaning | | 0 | Add | | 1 | Subtract | | 2 | Multiply | | 3 | Divide | | 4 | Exponent | | 5 | Root (reg. 0000 root reg. 0001) | |
| Do condition | cnd [ins] [command]  If reg. 0 [ins] reg. 1 then execute [command], [command] is in part 3 (except this), [ins] is 1 of the following:   |  |  | | --- | --- | | Binary code | Meaning | | 0 | Greater than | | 1 | Smaller than | | 2 | Not greater than | | 3 | Not smaller than | | 4 | Equal to | | 5 | Not equal to | |
| Do bitwise things (logic) | log [ins]  Do logic from reg. 0 and reg. 1 to reg. 2 (In the NOT case, reg. 0 is the only input, output is the same), [ins] is 1 of the following:   |  |  | | --- | --- | | Binary code | Meaning | | 0 | AND | | 1 | NOT | | 2 | NAND | | 3 | OR | | 4 | NOR | | 5 | XOR | | 6 | XNOR | |

1. Notes

* Any “0” and “1” on text must be “0” and “1”.
* Do not use “nested” IF.