TOY Reference Card

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Format 1	opcode				dest d				source s				source t			
Format 2	opcode				dest d				addr							

OpCode	Operation	Fmt	Pseudocode
0	halt	1	exit(0)
1	add	1	R[d] ← R[s] + R[t]
2	subtract	1	R[d] ← R[s] - R[t]
3	and	1	R[d] ← R[s] & R[t]
4	xor	1	R[d] ← R[s] ^ R[t]
5	shift left	1	R[d] ← R[s] << R[t]
6	shift right	1	R[d] ← R[s] >> R[t]
7	load addr	2	R[d] ← addr (i.e. load 1-byte constant)
8	load	2	R[d] ← mem[addr]
9	store	2	mem[addr] ← R[d]
A	load indirect	1	R[d] ← mem[R[t]]
B	store indirect	1	mem[R[t]] ← R[d]
C	branch zero	2	if (R[d] == 0) pc ← addr
D	branch positive	2	if (R[d] > 0) pc ← addr
E	jump register	2	pc ← R[d]
F	jump and link	2	R[d] ← pc; pc ← addr

Register 0 is always 0.
Loads from [FF] ← stdin.
Stores to [FF] → stdout.