Chimera-2018-A Emulator Assignment

Practical 4 - Inc and logic

CANS Tech INC

Implementing the INCA Instruction

Once again inside the Group_1 function switch add

```
case 0x38: // INCA
    CODE HERE
    break;
```

INCA		Addressing	Opcode
Increment Memory or Accumulator		A	0x38
Flags:	T - - - - - T -		
notes			

We simply increment Register A...

```
++Registers[REGISTER_A];
```

And check the N,Z flags...

```
set_flag_n(Registers[REGISTER_A]);
set_flag_z(Registers[REGISTER_A]);
```

Implementing the INX Instruction

Once again inside the Group_1 function switch add

```
case 0xA0: // INX
    CODE HERE
    break;
```

INX		Addressing	Opcode
Increments register X		impl	0xA0
Flags:	T - - - - - - -		
notes			

Just add...

++IndexRegister;

Don't forget about the flags as always, which is just the Z flag...

set_flag_z(IndexRegister);

Implementing the AND Instruction

Once again inside the Group_1 function switch add

```
case 0x50: // AND
    CODE HERE
    break;
```

AND		Addressing	Opcode
Register bitwise and with Accumulator		A-B	0x50
		A-C	0x51
Flags:	T - - - - - T -	A-D	0x52
notes		A-E	0x53
		A-F	0x54

Steps...

1. Copy your addiction op code
2. Replace the + with &
3. Remove adding the carry
4. Remove the code that sets the Carry flag
5. Remove the code that sets the Overflow flag
6. Add
   Flags = Flags & (0xFF - FLAG_V);
   to clear the overflow flag

Compile and run your code to see how many marks you have!

Now you can implement
AND, INC, DEX, INX, DEZ, INZ,

Now is a good time to catch up if you find yourself falling behind!