# Project Code.2.2

In this portion of the project you will transition to working in teams. You will be submitting 1 code base repository with your solution. To prepare for that you need to be aware of the high level design that your team is using because after this you will be using ONE of your implementations moving forward.

## Oh No. MORE New Opcodes!

You almost have a complete Turing Complete computer! Your computer is only missing a few opcodes:

- *Opcode 5* is `jump-if-true`: if the first parameter is non-zero, it sets the instruction pointer to the value from the second parameter. Otherwise, it does nothing.

- *Opcode 6* is `jump-if-false`: if the first parameter is zero, it sets the instruction pointer to the value from the second parameter. Otherwise, it does nothing.

- *Opcode 7* is `less than`: if the first parameter is less than the second parameter, it stores 1 in the position given by the third parameter. Otherwise, it stores 0.

- *Opcode 8* is `equals`: if the first parameter is equal to the second parameter, it stores 1 in the position given by the third parameter. Otherwise, it stores 0.

Like all instructions, these instructions need to support parameter modes as described previously in Part 2.1.

Normally, after an instruction is finished, the instruction pointer increases by the number of values in that instruction. However, if the instruction modifies the instruction pointer, that value is used and the instruction pointer is not automatically increased.

## Examples

For example, here are several programs that take one input, compare it to the value 8, and then produce one output:

| | |
|---|---|
| `3,9,8,9,10,9,4,9,99,-1,8` | Using position mode, consider whether the input is equal to 8; output 1 (if it is) or 0 (if it is not). |
| `3,9,7,9,10,9,4,9,99,-1,8` | Using position mode, consider whether the input is less than 8; output 1 (if it is) or 0 (if it is not). |
| `3,3,1108,-1,8,3,4,3,99` | Using immediate mode, consider whether the input is equal to 8; output 1 (if it is) or 0 (if it is not). |
| `3,3,1107,-1,8,3,4,3,99` | Using immediate mode, consider whether the input is less than 8; output 1 (if it is) or 0 (if it is not). |

## More Examples

Here are some jump tests that take an input, then output 0 if the input was zero or 1 if the input was non-zero:

| | |
|---|---|
| `3,12,6,12,15,1,13,14,13,4,13,99,-1,0,1,9` | (using position mode) |
| `3,3,1105,-1,9,1101,0,0,12,4,12,99,1` | (using immediate mode) |

Here's a larger example

```
3,21,1008,21,8,20,1005,20,22,107,8,21,20,1006,20,31,
1106,0,36,98,0,0,1002,21,125,20,4,20,1105,1,46,104,
999,1105,1,46,1101,1000,1,20,4,20,1105,1,46,98,99
```

The above example program uses an input instruction to ask for a single number. The program will then output 999 if the input value is below 8, output 1000 if the input value is equal to 8, or output 1001 if the input value is greater than 8.

The above example program uses an input instruction to ask for a single number. The program will then output 999 if the input value is below 8, output 1000 if the input value is equal to 8, or output 1001 if the input value is greater than 8.

## Challenge

This time, when the diagnostic program runs its input instruction to get the ID of the system to test, provide it 5. This diagnostic test suite only outputs one number, the diagnostic code.

What is the diagnostic code for system ID 5?

## Your Answer _____

Your program input is on the following page. It is the same input as Part 2.1