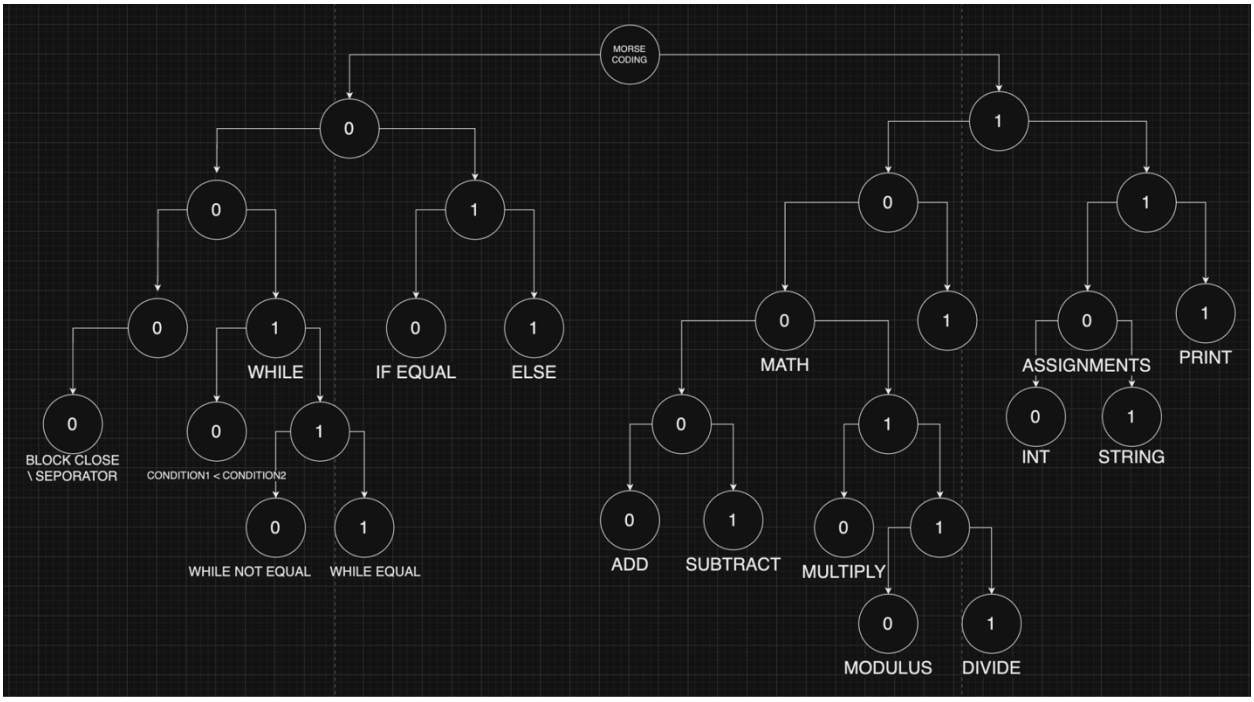
I HAVE NOT LEARNED WEB DEV – I WILL LEARN IT BEFORE I GRADUATE

MorseCoding - for people with only one key

Advantages
Only Requires one button - Thus it can be extremely useful on smaller or limited hardware.
The language is very lightweight and can be easily installed anywhere
Simple, in that it can be visualized with the following tree (next slide)

0's and 1's
The language is written in only 0's and 1's
This can be done with a traditional keyboard by pressing 0 and 1
Or, with only one key, a program will scan the key every second to determine whether a 0 or 1 should be written. Thus, it can be written with only 1 key.



Sample Programs

```
1101 010000
str  var

110110110110100 110100 110111110110100 110111110110100 111111111100
H               E      L               L               O
110111111100 111111111100 110111110100 110111110110100 111110110100 000
W            O            R            L               D

111 010000
pnt var
```

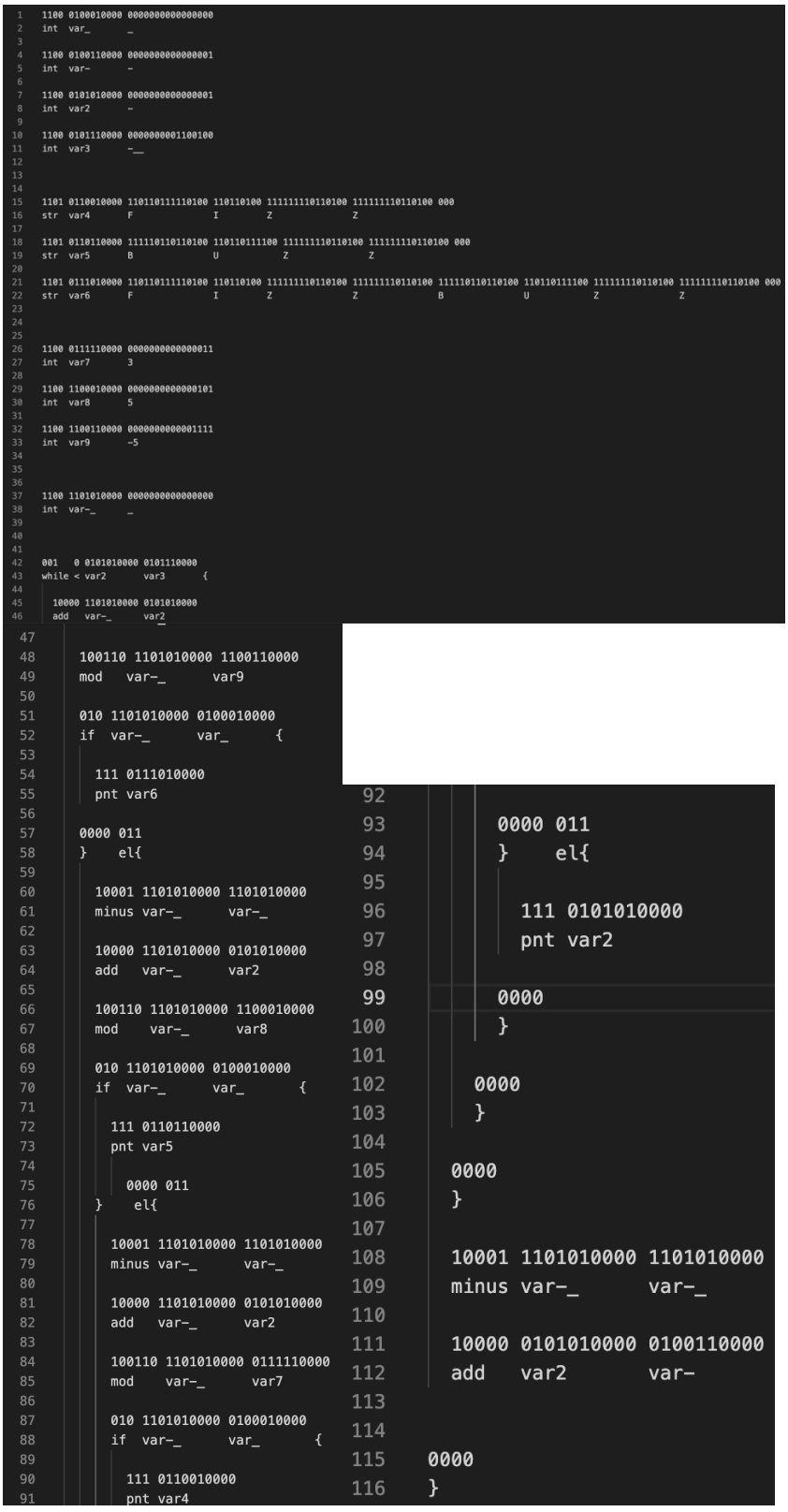
The words, letters, and spaces are not interpreted, they just help with readability
The program could be written like this. But it is far less clear what it does in this form

```
1101010000110110110110100110100110111110110100110111110110100111111111100
1101111111001111111111001101111101001101111101101001111101101000001110100000
```

Both these programs generate the same output
“HELLOWORLD”

```
1   1100 010000 0000000000000000
2   int  var    _
3
4   1100 110000 0000000000000001
5   int  var    -
6
7   1100 1110000 0000000000001010
8   int  var     -_
9
10  001   10 010000 1110000
11  while ne cond-  cond2
12  |
13  |  111 010000
14  |  pnt var
15  |
16  |  10000 010000 110000
17  |  add   var-   var2
18
19  0000
20  }
```

Here is an example of what loops look like in Morse Coding



```
1100 0100010000 0000000000000000
int  var_       _

1100 0100110000 0000000000000001
int  var-       -

1100 0101010000 0000000000000001
int  var2       -

1100 0101110000 0000000001100100
int  var3       -__



1101 0110010000 110110111110100 110110100 111111110110100 111111110110100 000
str  var4       F               I         Z               Z

1101 0110110000 111110110110100 110110111100 111111110110100 111111110110100 000
str  var5       B               U         Z               Z

1101 0111010000 110110111110100 110110100 111111110110100 111111110110100 1101101111100 111111110110100 111111110110100 000
str  var6       F               I         Z               Z               B               U           Z               Z



1100 0111110000 0000000000000011
int  var7       3

1100 1100010000 0000000000000101
int  var8       5

1100 1100110000 0000000000001111
int  var9       -5



1100 1101010000 0000000000000000
int  var-_      _



001   0 0101010000 0101110000
while < var2       var3      {

  10000 1101010000 0101010000
  add   var-_      var2

    100110 1101010000 1100110000
    mod    var-_      var9

    010 1101010000 0100010000
    if  var-_      var_      {

      111 0111010000
      pnt var6

    0000 011
    }    el{

      10001 1101010000 1101010000
      minus var-_      var-_

      10000 1101010000 0101010000
      add   var-_      var2

      100110 1101010000 1100010000
      mod    var-_      var8

      010 1101010000 0100010000
      if  var-_      var_       {

        111 0110110000
        pnt var5

          0000 011
      }    el{

        10001 1101010000 1101010000
        minus var-_      var-_

        10000 1101010000 0101010000
        add   var-_      var2

        100110 1101010000 0111110000
        mod    var-_      var7

        010 1101010000 0100010000
        if  var-_      var_       {

          111 0110010000
          pnt var4

            0000 011
            }    el{

              111 0101010000
              pnt var2

            0000
            }

      0000
      }

  0000
  }

  10001 1101010000 1101010000
  minus var-_      var-_

  10000 0101010000 0100110000
  add   var2       var-


0000
}
```

FIZZ BUZZ