MorseCoding

for people with only one key

Advantages

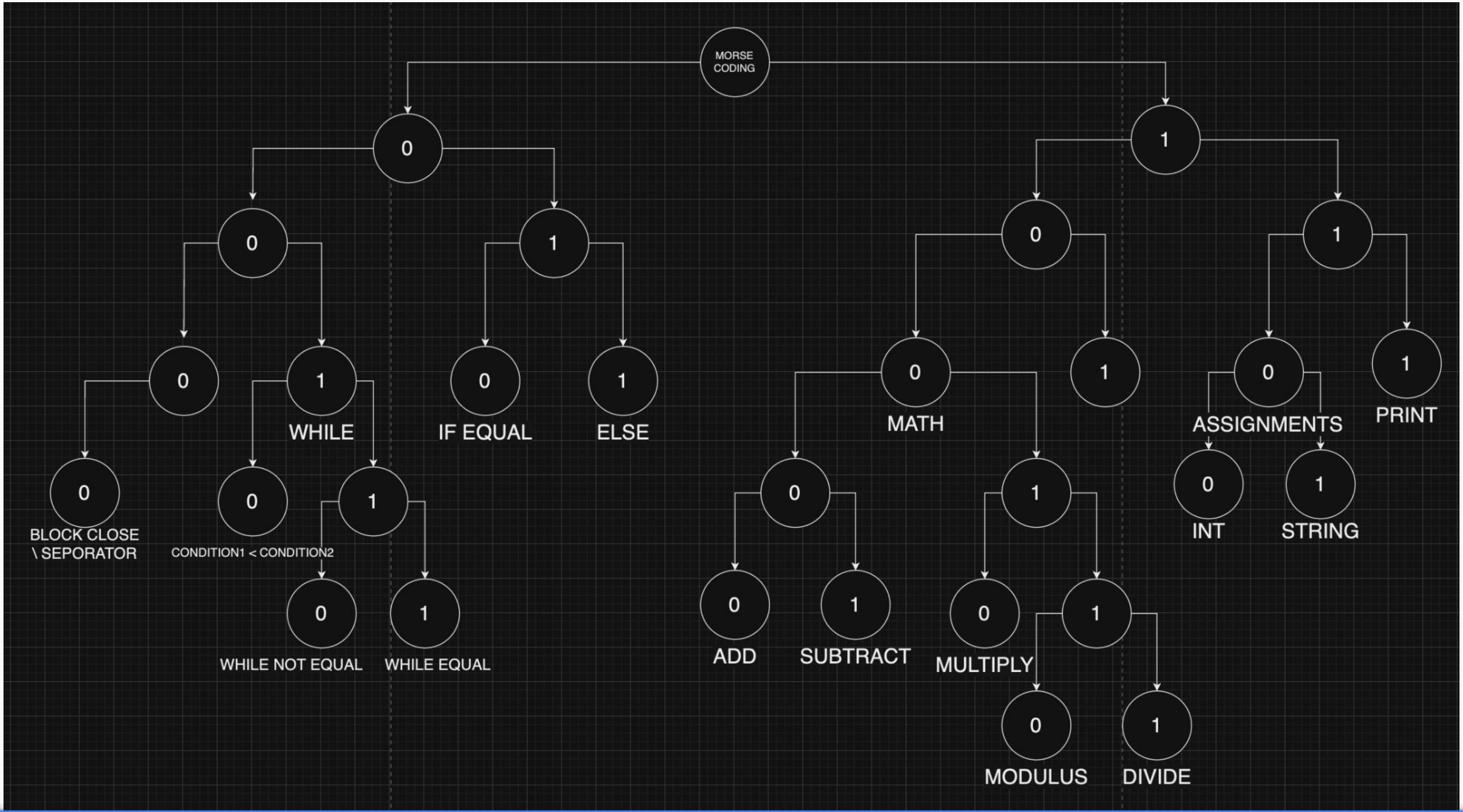
- Only Requires one button - Thus it can be extremely useful on smaller or limited hardware.
- The language is very lightweight and can be easily installed anywhere
- Simple, in that it can be visualized with the following tree (next slide)

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.



MorseCodingTree

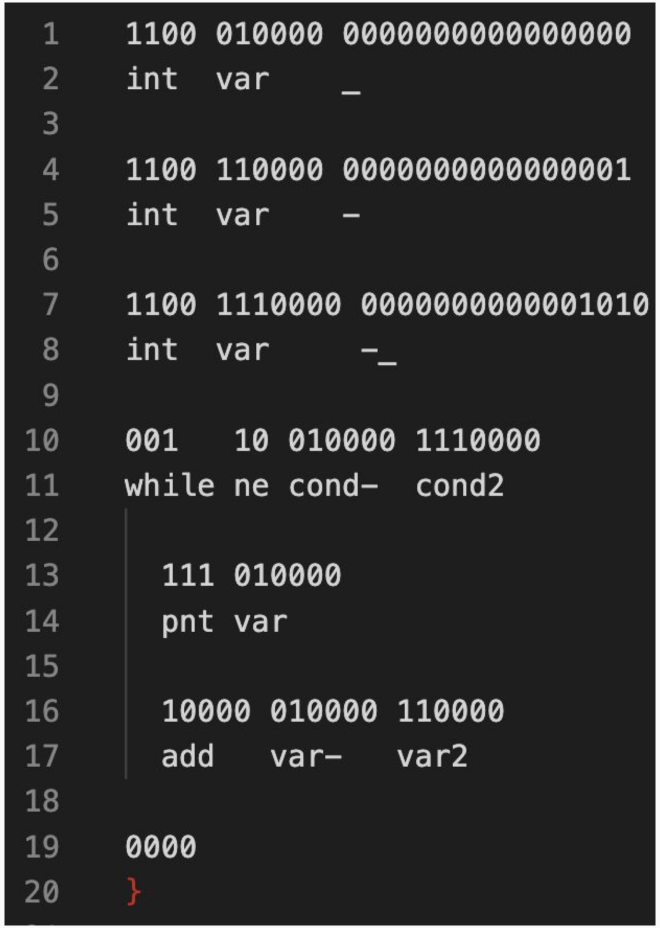
```
1101 010000
str  var

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

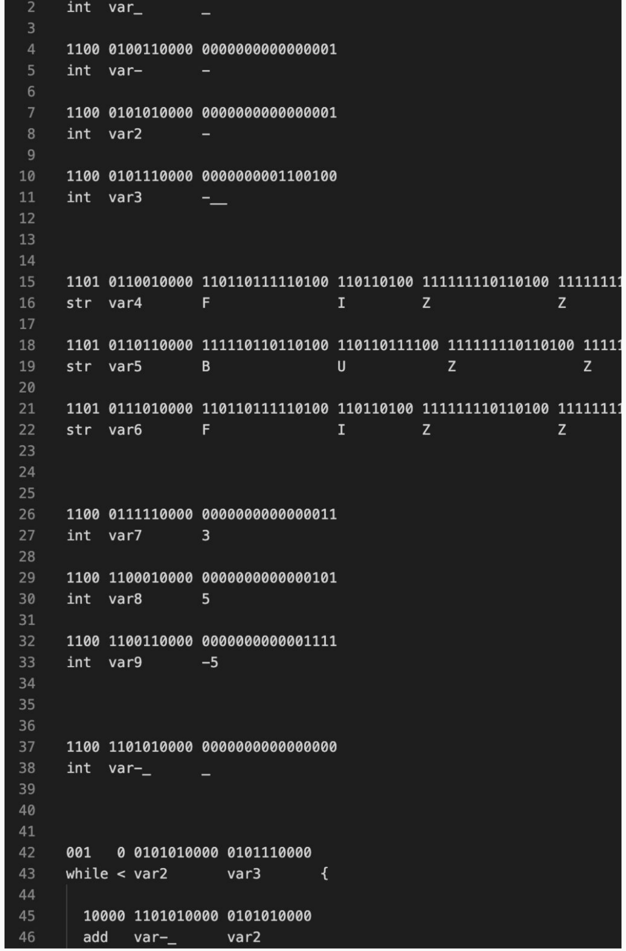
Both these programs generate the same output "HELLOWORLD"

```
1101010000110110110110100110100110111110110100110111110110100111111111100
110111111100111111111100110111110100110111110110100111110110100000111010000
```

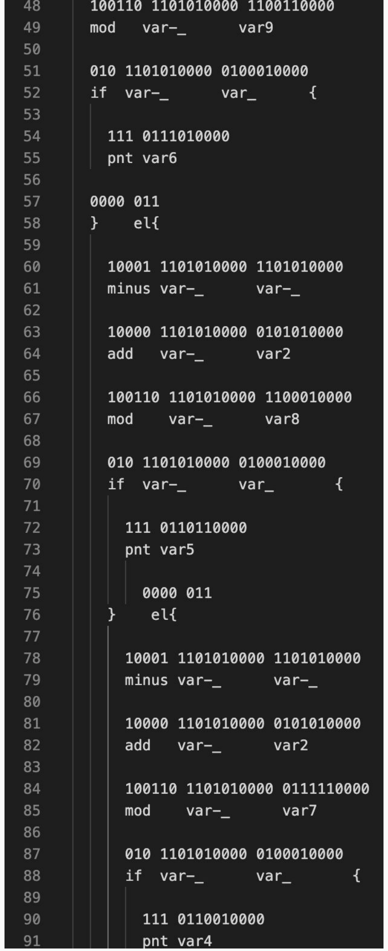


```
1100 010000 0000000000000000
int  var    _

1100 110000 0000000000000001
int  var    –

1100 1110000 0000000000001010
int  var     –_

001   10 010000 1110000
while ne cond–  cond2

  111 010000
  pnt var

  10000 010000 110000
  add   var–   var2

0000
}
```

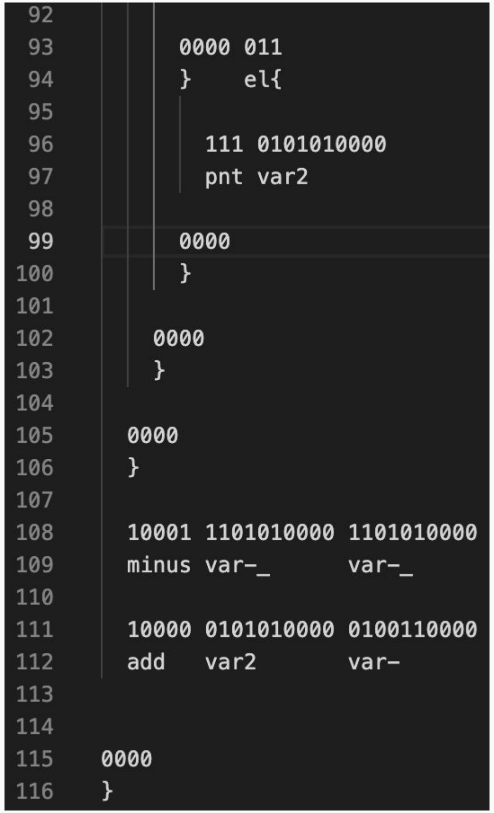
Here is an example of what loops look like in Morse Coding



```
int  var_       _

1100 0100110000 0000000000000001
int  var-       -

1100 0101010000 0000000000000001
int  var2       -

1100 0101110000 0000000001100100
int  var3       -__



1101 0110010000 110110111110100 110110100 111111110110100 11111111
str  var4       F               I         Z               Z

1101 0110110000 111110110110100 110110111100 111111110110100 11111
str  var5       B               U            Z               Z

1101 0111010000 110110111110100 110110100 111111110110100 11111111
str  var6       F               I         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
}
```

FIZZBUZZ

```
1100010001000000000000000000001100010011000000000000000000011100
0101010000000000000000000111000101110000000000000110010011010110
0100001101101111101001101101001111111101101001111111101101000001
1010110110000111110110110100110110111100111111110110100111111110
1101000001101011101000011011011111010011011010011111111011010011
1111110110100111110110110100110110111100111111110110100111111110
1101000001100011111000000000000000000011110011000100000000000000
0010111001100110000000000000000111111100110101000000000000000000
0001001010100000101110000100001101010000000101000010011011010100
0011001100000101101010000010001000011101110100000000011100011101
0100001101010000100001101010000010101000010011011010100001100010
0000101101010000010001000011101101100000000011100011101010000110
1010000100001101010000010101000010011011010100000111110000010110
1010000010001000011101100100000000011111010101000000000000000010
0011101010000110101000010000010101000000100110000000
```

This is also FizzBuzz