An Introduction to STEM Programming with Python 3 – Chapter 2
Other Bases - Binary

By
James M. Reneau, Ph.D.
jreneau@shawnee.edu

Shawnee State University

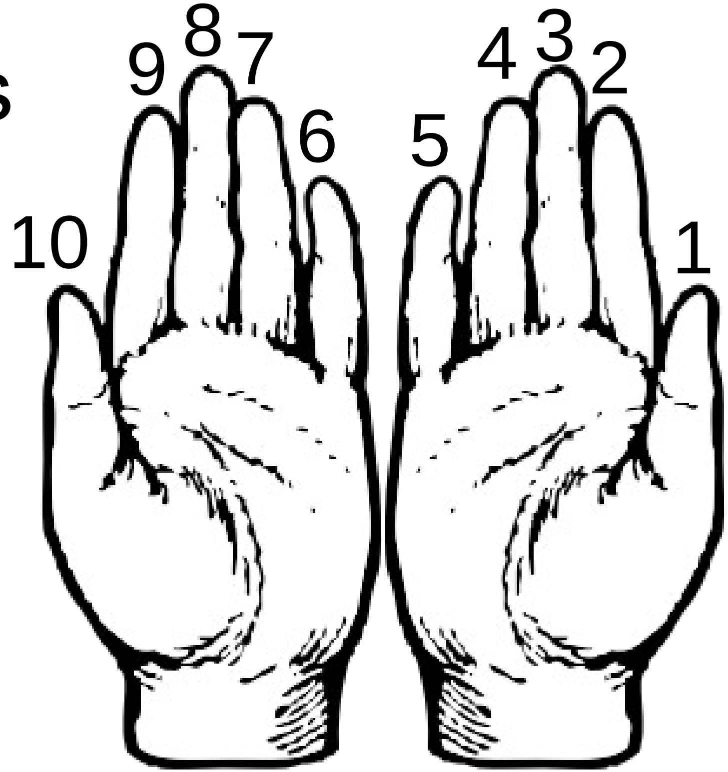
Version 2019-12-12a

In this video we will cover:

- Decimal numbers (BASE 10)
- The positional notation for representing numbers
- Binary numbers (BASE 2)
- The remainder method for converting decimal to binary.
- The positional method for converting binary to decimal
- Binary numbers in Python

Decimal Numbers

- Decimal numbers use a base of 10.
- We write large numbers using a positional notation.
  - For numbers on the left of the decimal point we increase the power of 10
  - For numbers on the right of the decimal point we decrease the power of 10



Decimal Numbers

- Using the positional notation:
- 123.45 can be written as

  $1x10^2 + 2x10^1 + 3x10^0 + 4x10^{-1} + 5x10^{-2}$
- And 987654321 can be written as
  $9\ x10^8 + 8x10^7 + 7x10^6 + 6x10^5 + 5x10^4$
  $+ 4x10^3 + 3x10^2 + 2x10^1 + 1x10^0$

Binary – Base 2

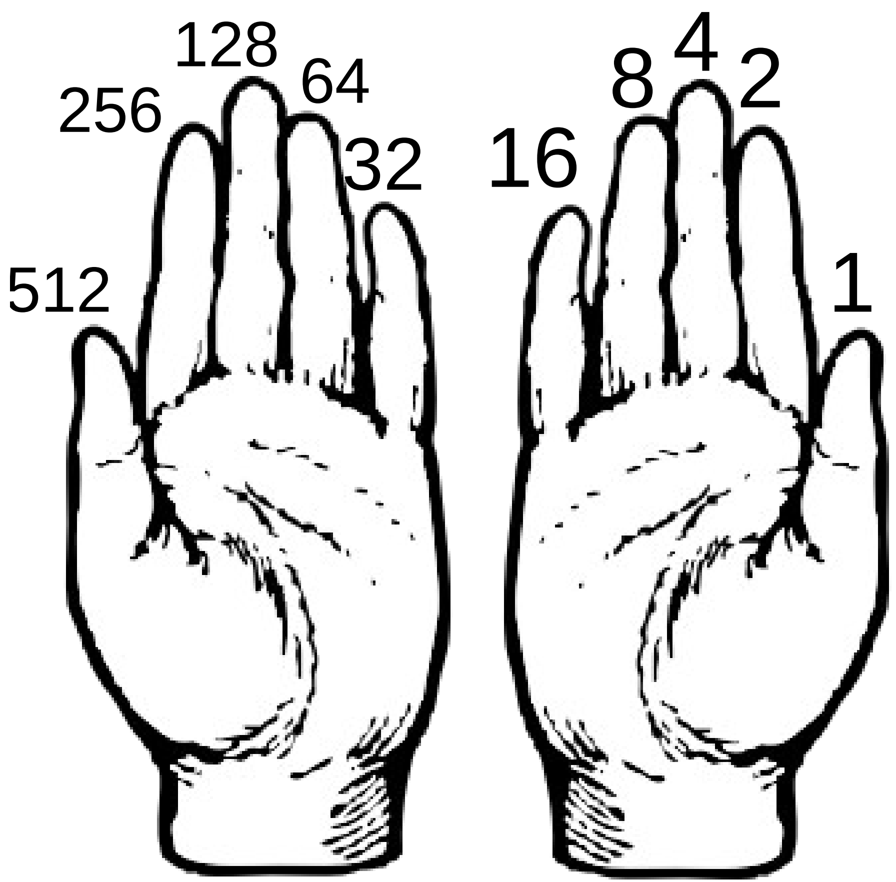
- The computer was born without hands but it has switches that are
  - either on or off – 1 or 0
- If you can only count using a positional sequence of 1 or 0 we would call that base 2 – Binary
- Instead of using powers of 10 the computer represents numbers using powers of 2

Binary Numbers

- Using the positional notation for understanding and converting binary back to decimal:
- 10110_2 can be thought of as
- $1x2^4 + 0x2^3 + 1x2^2 + 1x2^1 + 0x2^0$
  $16 + 4 + 2$
  22_{10}
- And 111101_2 can be written as
  $1x2^5 + 1x2^4 + 1x2^3 + 1x2^2 + 0x2^1 + 1x2^0$
  $32 + 16 + 8 + 4 + 1$
  61_{10}

Powers of 2

- $2^0 = 1$
- $2^1 = 2$
- $2^2 = 4$
- $2^2 = 8$
- $2^4 = 16$
- $2^5 = 32$
- $2^6 = 64$
- $2^7 = 128$
- $2^8 = 256$
- $2^9 = 512$
- $2^{10} = 1024$
- $2^{11} = 2048$
- $2^{12} = 4096$
- $2^{13} = 8192$
- $2^{14} = 16384$
- $2^{15} = 32768$



Converting Decimal to Binary Remainder Method

- Convert 101_{10} to binary

```
101 / 2 = 50 r 1
 50 / 2 = 25 r 0
 25 / 2 = 12 r 1
 12 / 2 =  6 r 0
  6 / 2 =  3 r 0
  3 / 2 =  1 r 1
  1 / 2 =  0 r 1
```

- Write the remainders in the reverse order you calculated them
- 1100101_2

Binary in Python

- Binary literal
  - Prefix the binary number with "`0b`"
- Convert decimal integer to binary
  - `bin( number )`
  - Function takes integer and returns a string (with the 0b)

Binary in Python

```python
# binary literal
a = 0b1011010
print(a)

# convert decimal to binary string
x = 98989
print(bin(x))
```

```
90
0b11000001010101101
```

Thank you

2. Contact me at jreneau@shawnee.edu