Comp. Logic HW 1

Andrew Householder

September 9, 2021

Problems 6, 8, 10, 13, 15

converter.py

```
def dec_to_bin(decimal):
    binary = 0
    weight = 0

    while decimal > 0:
        bit = (decimal % 2) *10**weight
        binary += bit
        weight += 1
        decimal = decimal // 2
    return binary

def bin_to_dec(binary):
    weight = 0
    decimal = 0
    remainder = binary % 10

    while binary > 0:
        binary = binary // 10
        decimal += remainder*(2**weight)
        remainder = binary % 10
        weight += 1
    return decimal

def print_largest_dec(digits):
    print(2**digits - 1)

def make_bin_sequence(minimum, maximum):
    for decimal in range(minimum, maximum + 1):
        print(f"Decimal: {decimal}", end=" ")
        print(f"Binary: {dec_to_bin(decimal)}")
```

```
def bin_addition(a, b):
    sum = bin_to_dec(a) + bin_to_dec(b)
    return dec_to_bin(sum)
```

solutions.py

```
import converter

#problem 6
print("2-2 #6")
bins = [1110, 1010, 111000, 10000, 10101, 11101, 10111, 11111]

for binary in bins:
    print(f"Binary: {binary}", end=" ")
    print(f"Decimal: {converter.bin_to_dec(binary)}")

print("\n")

#problem 8
print("2-2 #8")
for digit in range(2, 12):
    converter.print_largest_dec(digit)

print("\n")

#problem 10
print("2-2 #10")
converter.make_bin_sequence(0,7)
converter.make_bin_sequence(8,15)
converter.make_bin_sequence(16,31)
converter.make_bin_sequence(32,63)
converter.make_bin_sequence(64,75)

print("\n")

#problem 13
print("2-3 #13")
decs = [15, 21, 28, 34, 40, 59, 65, 73]
for decimal in decs:
    print(f"Decimal: {decimal}")
    print(f"Binary: {converter.dec_to_bin(decimal)}")

print("\n")

#problem 15
```

```
print("2-4 #15")

print(converter.bin_addition(11, 1))
print(converter.bin_addition(10, 10))
print(converter.bin_addition(101, 11))
print(converter.bin_addition(111, 110))
print(converter.bin_addition(1001, 101))
print(converter.bin_addition(1101, 1011))

print("\n")
```

2-5

19

1111, 1111 1111

2-6

28

a.

start: 1001 1001
invert: 0110 0110
add 1: 0110 0111

b.

start: 0111 0100
invert: 1000 1011
add 1: 1000 1100

c.

start: 1011 1111
invert: 0100 0000
add 1: 0100 0001

29

a.

$$0111110000101011$$

$$01.11110000101011 \times 2^{14}$$

$$14 + 127 = 141$$

0	1000 1101	1111 0000 1010 1100 0000 000
S	Exponent	Mantissa

b.

$$100110000011000$$

$$1.10000011000 \times 2^{11}$$

$$11 + 127 = 138$$

1	1000 1010	1000 0011 0000 0000 0000 000
S	Exponent	Mantissa