CMPT295: Assignment 2

Daniel Wallman (301383524)

January 24, 2019

1. (a) See tables.

Q	R
106	-
53	0
26	1
13	0
6	1
3	0
1	0
0	1

$= 0110\ 1010$

Q	R
128	-
64	0
32	0
16	0
8	0
4	0
2	0
1	0
0	1

$= 1000\ 0000$

Q	R
150	-
75	0
37	1
18	1
9	0
4	1
2	0
1	0
0	1

$= 1001\ 0110$

```
# 106 (hex)
0110 = 6, 1010 = A
= 6A

# 128 (hex)
1000 = 8, 0000 = 0
= 80

# 150 (hex)
1001 = 9, 0110 = 6
= 96
```

(b)
```
# -1
0000 0001
1111 1110    # invert
1111 1111    # add 1

# -106
0110 1010
1001 0101
1001 0110
```

```
# -128
1000 0000
0111 1111
1000 0000
```

(c) 1010 1110 (unsigned) = 174
    1010 1110 (2's comp) = -82

(d) # first
    1100 = C, 1110 = E
    = CE
    0011 = 3, 0111 = 7
    = 37

    # second
    1111 = F, 1010 = A
    = FA
    1010 = A, 1110 = E
    = AE

$$\begin{array}{r} \overset{1}{c}e \\ +\ \ 37 \\ \hline 105 \end{array}$$

$$\begin{array}{r} \overset{1}{f}a \\ +\ \ ae \\ \hline 1a8 \end{array}$$

2. (a) 1,2,3,5,9

   (b) #13
       lea(%edi,%edi,2),%eax
       lea(%edi,%eax,4),%edi

       #20
       mov $0, %eax
       lea(%edi,%edi,4),%edi
       lea(%eax,%edi,4),%edi

       #37
       lea(%edi,%edi,8),%eax
       lea(%edi,%eax,4),%edi

3. Variables:
   eax = result
   edi = x

```
esi = kth bit
edx = squared-test
ecx = counter
```

Square root through digit-by-digit algorithm.

```
7          .globl sqrt
8  sqrt:
9          movl $0, %eax
10         movl $15, %ecx
11 loop:
12         cmpl $0, %ecx
13         jl end
14
15         movl $1, %esi          # set kth bit to 1
16         shll %cl, %esi
17         orl %esi, %eax
18
19         mov %eax, %edx         # if x*x > x set kth bit to 0
20         imull %edx, %edx
21         cmpl %edx, %edi
22         jl undo
23         decl %ecx              # else iterate again
24         jmp loop
25 undo:
26         xorl %esi, %eax        # set kth bit to 0
27         decl %ecx
28         jmp loop
29 end:
30         ret
```