CNS Lab Assignment 3: Tools

111703013 Akshay Deodhar

Code

1 GCD

```python

from sys import argv, stderr

def gcd(a, b):
    small = min(a, b)
    big = max(a, b)

    if small == 0:
        return big

    return gcd(big % small, small)


if __name__ == "__main__":

    n = len(argv)

    if n != 3:
        print("usage: python3 gcd.py <num1> <num2>", file = stderr)
        exit(1)


    __, a, b = argv
    a = int(a)
    b = int(b)

    print("GCD({0}, {1}) = {2} ".format(a, b, gcd(a, b)))
```

2 Modulo

```python



from sys import argv, stderr


if __name__ == "__main__":

    n = len(argv)

    if n != 3:
        print("usage: python3 mod.py <num1> <num2>", file = stderr)
        exit(1)

```

```
15
16      __, a, b = argv
17      a = int(a)
18      b = int(b)
19
20      print("{0} % {1} = {2}".format(a, b, a % b))
```

3 Shift

```
1
2   from sys import argv, stderr
3
4
5   if __name__ == "__main__":
6
7       n = len(argv)
8
9       if n != 4:
10          print("usage: python3 mod.py <num1> <dir> <bits>", file = stderr)
11          exit(1)
12
13
14      __, n, direction, bits = argv
15      n = int(n)
16      bits = int(bits)
17
18      if direction not in ['L', 'R']:
19          print("Invalid direction {}".format(direction))
20
21
22      if direction == 'L':
23          print("{0} << {1} = {2}".format(n, bits, n << bits))
24      else:
25          print("{0} >> {1} = {2}".format(n, bits, n >> bits))
```

4 XOR

```
1
2
3   from sys import argv, stderr
4
5
6   if __name__ == "__main__":
7
8       n = len(argv)
9
10      if n != 3:
11          print("usage: python3 xor.py <num1> <num2>", file = stderr)
12          exit(1)
13
14
15      __, a, b = argv
16      a = int(a)
17      b = int(b)
18
19      print("{0} ^ {1} = {2}".format(a, b, a ^ b))
```

Output

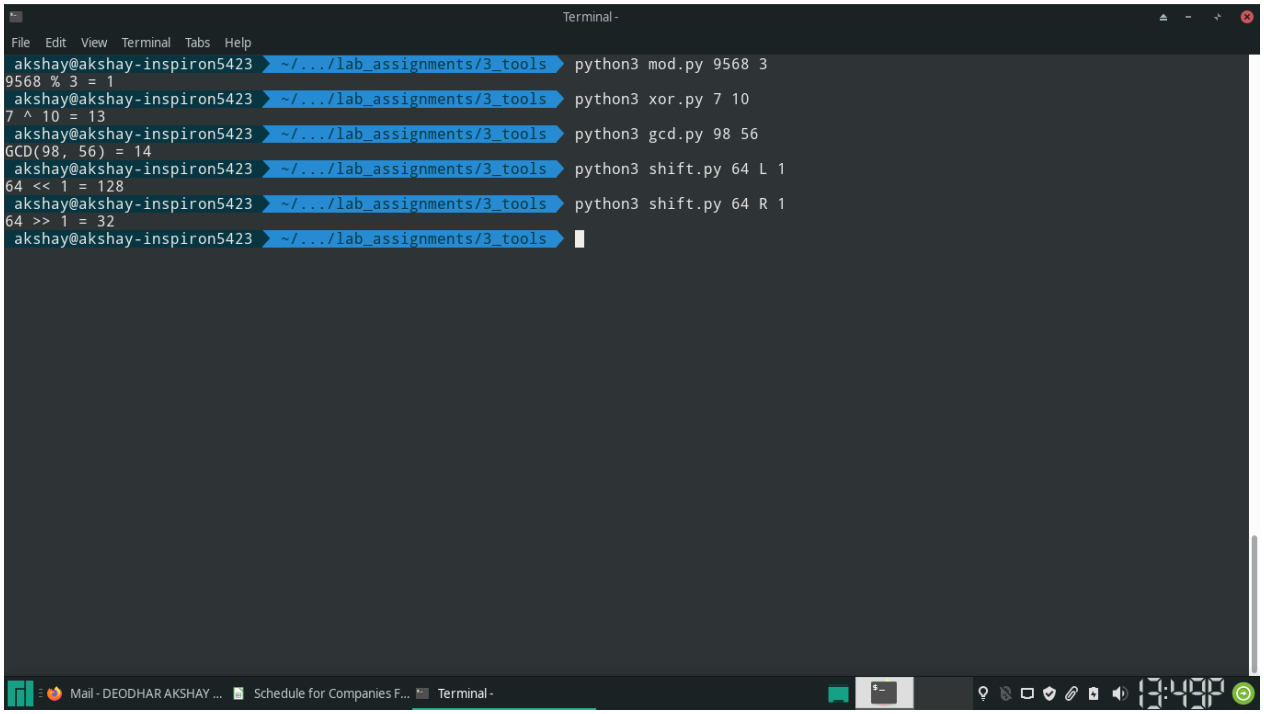

Figure 1: Output of execution of each of the functions, from the command line