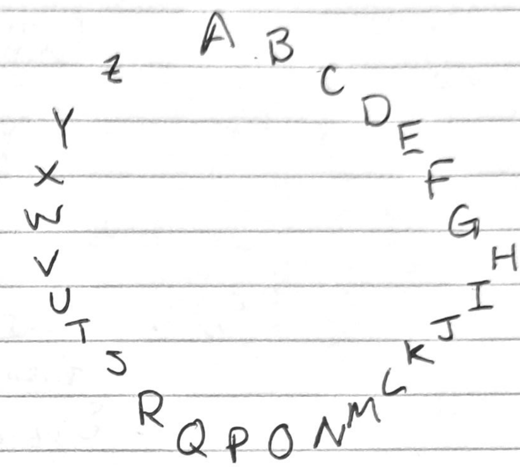
Math 470

Cole McAnelly

Homework 1

1) Decode the Caesar Cipher



When $k = 22 = -4$

$\therefore$ "ANAGG" probably "Aggie" (?)

$\Rightarrow$ "An Aggie does not lie cheat or Steal or tolerate those who do"

2) 1.3a $\rightarrow$ Encrypt the plaintext message

"The gold is hidden in the garden"

$\hookrightarrow$ IBX FEPA QL BQAAXW QW IBX FSVAXW

3) 1.6b $\rightarrow$ Let $a, b, c \in \mathbb{Z}$. Use the def. of divisibility to prove the following prop of divisibility

If $a|b$ and $b|a$, then $a = \pm b$

$b = ax$ & $a = by \Rightarrow \cancel{x} y = \frac{a}{b}$

$\hookrightarrow x = \frac{b}{a}$

- Since x is an integer $|b| \leq |a|$
- Since y is an integer $|a| \leq |b|$

$\therefore |a| = |b| \Rightarrow a = \pm b$

4) 1.9a → Use Euclidean algorithm to compute gcd

$\gcd(291, 252) = 3$

$$291 = 1 \cdot 252 + 39$$
$$252 = 6 \cdot 39 + 18$$
$$39 = 2 \cdot 18 + 3$$
$$18 = 6 \cdot 3 + 0$$

3

5) 1.11 Let a & b be positive integers

a) Given $u, v \in \mathbb{Z}$: $au + bv = 1$. Prove that $\gcd(a,b) = 1$

b) Suppose that there are ints u & v $au + bv = 6$. Is $\gcd(a,b) = 6$? If not, give a counterexample.

a) According to the Extended Euclidean Algorithm when computing the $GCD(a,b)$, we compute coefficients $u, v \in \mathbb{Z}$, such that $au + bv = \gcd(a,b)$.

$\therefore$ because $au + bv = 1 \Leftrightarrow \gcd(a,b) = 1$

By definition a, b are coprime. (aka Relatively Prime)

b) No it is not necessarily true, Counterexample:

Take $a = 2$, $b = 4$ and $u, v = 1 \Rightarrow (2 \cdot 1) + (4 \cdot 1) = 6$

However $\gcd(2,4) \neq 6$.

General:

All the possible values of $\gcd(a,b)$ are $\{1, 2, 3, 6\}$.

This can be found by taking the following process

$$\gcd(a,b) \mid a \text{ AND } \gcd(a,b) \mid b \Rightarrow \gcd(a,b) \mid \underbrace{(au + bv)}_{=6}$$

$\Rightarrow \gcd(a,b) \mid 6$

$\therefore \{d \in \mathbb{Z}^+ : d \mid 6\}$

gcd.py > main

```python
def gcd(a:int, b:int) -> int:
    if (b == 0): return a               # Base Case: When one arg is 0, the other is the gcd
    if (b > a): return gcd(b, a)        # Swap Case: When `b` is larger than `a`, swap the args
    return gcd(b, a % b)                # Recursive: Else, return gcd of `b` and the remainder of `a/b`


def main() -> None:
    a = 1234567890123456789012345678901234567890123456789012345678901234567890123456789
    b = 234567890123456789012345678901234567890123456789012345678901234567890123456789
    print("Given:\n\n\ta = %d\n\tb = %d\n\nGCD(a,b) = %d" % (a, b, gcd(a,b)) )
    print("\nWhere:")
    print("\ngcd(a:int, b:int) -> int:")
    print("\tif b == 0 return a")
    print("\tif b > a return gcd(b,a)")
    print("\treturn gcd(b, a rem b)")


if __name__ == "__main__": main()
```

PROBLEMS OUTPUT TERMINAL PORTS 1 DEBUG CONSOLE

```
✓ 13:32:50 colemcanelly ➜ [hw1]
$ python3.11 gcd.py
Given:

        a = 1234567890123456789012345678901234567890123456789012345678901234567890123456789
        b = 234567890123456789012345678901234567890123456789012345678901234567890123456789

GCD(a,b) = 1

Where:

gcd(a:int, b:int) -> int:
        if b == 0 return a
        if b > a return gcd(b,a)
        return gcd(b, a rem b)
```