Homework-3 Report

Emre Can Tüzer

BGK-516

Enver ÖZDEMİR

1)

Before writing the code, I wanted to understand the necessary steps mathematically and did research on the relevant equation.

$$a \cdot x + b \cdot y = gcd(a, b)$$

We can write $a = bq + r$, we observe that

$$gcd(a, b) = gcd(b, r)$$

Also we can write this equation

$$\mathrm{b} \cdot x_1 + \mathrm{r} \cdot y_1 = \mathrm{gcd(b, r)}$$

Also we know that $a - bq = r$

$$\mathrm{b} \cdot x_1 + (\mathrm{a} - \mathrm{bq}) \cdot y_1 = \mathrm{gcd(a, b)}$$

$$\mathrm{a} \cdot y_1 + \mathrm{b} \cdot (x_1 - \mathrm{q} \cdot y_1) = \mathrm{gcd(a, b)}$$

Thanks to this equation, we can start our coding process.

$$\mathrm{x} = y_1$$

$$\mathrm{y} = x_1 - \mathrm{q} \cdot y_1$$

I used a recursive function in my code, the reason for this comes from the basic logic of the Extended Euclidean Algorithm and continues until r = 0. When r = 0, The operation of the function ends and this returns $\boldsymbol{gcd}$ and the coefficients $\boldsymbol{x}, \boldsymbol{y}$.

$$a = bq + r$$

```
void extended_gcd(mpz_t gcd, mpz_t x, mpz_t y, mpz_t a, mpz_t b) {
    if (mpz_cmp_ui(b, 0) == 0) {
        mpz_set(gcd, a);
        mpz_set_ui(x, 1); // x = 1
        mpz_set_ui(y, 0); // y = 0
        return;
    }
    if (mpz_cmp_ui(a, 0) == 0) {
        printf( format: "Error: a cannot be zero.\n");
        mpz_set_ui(gcd, 0);  // gcd = 0
        mpz_set_ui(x, 0);    // x = 0
        mpz_set_ui(y, 0);    // y = 0
        return;
    }
    mpz_t x1, y1, r;
    mpz_inits(x1, y1, r, NULL);
    mpz_mod(r, a, b);

    extended_gcd(gcd, x1, y1,  a: b,  b: r);
```

You can see sample outputs below. I developed this assignment using the gmp library to obtain accurate results at high a and b values.

```
mpz_set_str(a, "6565465468664", 10);
mpz_set_str(b, "1945418", 10);
```

```
C:\Users\emrecan\Desktop\BGK_516_HW3\question1\cmake-build-debug\question1.exe
a = 6565465468664
b = 1945418
x = -161094
y = 543665728501
GCD(6565465468664, 1945418) = 2
Process finished with exit code 0
```

```
mpz_set_str(a, "64221646564564426426426564 65", 10);
mpz_set_str(b, "455245445465645646546452135646462426 4", 10);
```

```
C:\Users\emrecan\Desktop\BGK_516_HW3\question1\cmake-build-debug\question1.exe
a = 64221646564564426426426564 65
b = 455245445465645646546452135646462426 4
x = 492557255827001186326795615369503265
y = -69485237714301377814104876 6
GCD(64221646564564426426426564 65, 455245445465645646546452135646462426 4) = 1
Process finished with exit code 0
```

```
mpz_set_str(a, "642216465644564564426426426564 65", 10);
mpz_set_str(b, "0", 10);
```

```
C:\Users\emrecan\Desktop\BGK_516_HW3\question1\cmake-build-debug\question1.exe
a = 642216465644564564426426426564 65
b = 0
x = 1
y = 0
GCD(642216465644564564426426426564 65, 0) = 642216465644564564426426426564 65
Process finished with exit code 0
```

```
mpz_set_str(a, "0", 10);
mpz_set_str(b, "455245445465645646546452135646462426 4", 10);
```

```
C:\Users\emrecan\Desktop\BGK_516_HW3\question1\cmake-build-debug\question1.exe
Error: a cannot be zero.
a = 0
b = 455245445465645646546452135646462426 4
x = 0
y = 0
GCD(0, 455245445465645646546452135646462426 4) = 0
Process finished with exit code 0
```

2.a)

A multiplication group's order, mod n, indicates how many prime elements there are between them. Using Euler's Totient Function $\boldsymbol{\phi(n)}$, we may compute this:

$$\boldsymbol{\phi(n) = n \cdot \prod_{p|n} \left(1 - \frac{1}{p}\right)}$$

$\boldsymbol{G_1 = Z_{667}}$: $667 = 23 \cdot 29$

$$\phi(667) = 667 \cdot (1 - \frac{1}{23}) \cdot (1 - \frac{1}{29})$$

$$\phi(667) = 667 \cdot \frac{22}{23} \cdot \frac{28}{29} = 667 \cdot 0.9565 \cdot 0.9655 \approx \mathbf{616}$$

$\boldsymbol{G_2 = Z_{12857}}$: $12857 = 13^2 \cdot 761$

$$\phi(12857) = 12857 \cdot (1 - \frac{1}{13}) \cdot (1 - \frac{1}{761})$$

$$\phi(12857) = 12857 \cdot \frac{12}{13} \cdot \frac{760}{761} = \mathbf{11088}$$

$\boldsymbol{G_3 = Z_{131461}}$: $131461 \cdot 1$

$$\phi(131461) = 131461 - 1 = \mathbf{131460}$$

$\boldsymbol{H_1 = Z_{673}}$: $673 \cdot 1$

$$\phi(673) = 673 - 1 = \mathbf{672}$$

$\boldsymbol{H_2 = Z_{12889}}$: $12889 \cdot 1$

$$\phi(12889) = 12889 - 1 = \mathbf{12888}$$

$\boldsymbol{H_3 = Z_{131477}}$: $131477 \cdot 1$

$$\phi(131477) = 131477 - 1 = \mathbf{131476}$$

2.b)

To find the rank of an element, we do the following, where k is the rank of the element and k must be a number that exactly divides the total rank of the group $\boldsymbol{\phi(n)}$.

$$\mathbf{a}^{\boldsymbol{k}} \equiv \mathbf{1} \pmod{\mathbf{n}}$$

$\boldsymbol{G_1 = Z_{667}}$ $\boldsymbol{order\ 616}$:

For **a = 2**

$$2^1 \equiv 2 \pmod{667}$$

$$2^2 \equiv 4 \pmod{667}$$

$$2^3 \equiv 8 \pmod{667}$$

.

.

$$2^{616} \equiv 1 \pmod{667}$$

For **a = 5**

$$5^1 \equiv 5 \pmod{667}$$

$$5^2 \equiv 25 \pmod{667}$$

$$5^3 \equiv 125 \pmod{667}$$

.

.

$$5^{308} \equiv 1 \pmod{667}$$

$\boldsymbol{G_2 = Z_{12857}}$ $order\ 11088$:

For **a = 2**

$$2^1 \equiv 2 \pmod{12857}$$

$$2^2 \equiv 4 \pmod{12857}$$

$$2^3 \equiv 8 \pmod{12857}$$

.

.

$$2^{11088} \equiv 1 \pmod{12857}$$

For **a = 3**

$$5^1 \equiv 3 \pmod{12857}$$

$$5^2 \equiv 9 \pmod{12857}$$

$$5^3 \equiv 27 \pmod{12857}$$

.

.

$$5^{2772} \equiv 1 \pmod{12857}$$

$\boldsymbol{G_3 = Z_{131461}}$ $order$ 131460:

For **a = 2**

$$2^1 \equiv 2 \pmod{131461}$$

$$2^2 \equiv 4 \pmod{131461}$$

$$2^3 \equiv 8 \pmod{131461}$$

.

.

$$2^{11088} \equiv 1 \pmod{131461}$$

For **a = 3**

$$3^1 \equiv 3 \pmod{131461}$$

$$3^2 \equiv 9 \pmod{131461}$$

$$3^3 \equiv 27 \pmod{131461}$$

.

.

$$3^{131460} \equiv 1 \pmod{131461}$$

$\boldsymbol{H_1 = Z_{673}}$ $order$ 672:

For **a = 2**

$$2^1 \equiv 2 \pmod{673}$$

$$2^2 \equiv 4 \pmod{673}$$

$$2^3 \equiv 8 \pmod{673}$$

.

.

$$2^{672} \equiv 1 \pmod{673}$$

For **a = 3**

$$3^1 \equiv 3 \pmod{673}$$

$$3^2 \equiv 9 \pmod{673}$$

$$3^3 \equiv 27 \pmod{673}$$

.

.

$$3^{672} \equiv 1 \pmod{673}$$

$\boldsymbol{H_2 = Z_{12889}}$ $order$ 12888:

For **a = 2**

$$2^1 \equiv 2 \pmod{12889}$$

$$2^2 \equiv 4 \pmod{12889}$$

$$2^3 \equiv 8 \pmod{12889}$$

.

.

$$2^{12888} \equiv 1 \pmod{12889}$$

For **a = 3**

$$3^1 \equiv 3 \pmod{12889}$$

$$3^2 \equiv 9 \pmod{12889}$$

$$3^3 \equiv 27 \pmod{12889}$$

.

.

$$3^{12888} \equiv 1 \pmod{12889}$$

$\boldsymbol{H_3} = \boldsymbol{Z}_{131477}$ $order$ 131476:

For **a = 2**

$$2^1 \equiv 2 \pmod{131477}$$

$$2^2 \equiv 4 \pmod{131477}$$

$$2^3 \equiv 8 \pmod{131477}$$

.

.

$$2^{131476} \equiv 1 \pmod{131477}$$

For **a = 3**

$$3^1 \equiv 3 \pmod{131477}$$

$$3^2 \equiv 9 \pmod{131477}$$

$$3^3 \equiv 27 \pmod{131477}$$

.

.

$$3^{131476} \equiv 1 \pmod{131477}$$

2.c)

Elements (primitive roots) having an order equal to the total order $\boldsymbol{\phi}(\boldsymbol{n})$ are required if the group is evolutionary. The equality of the total divided equal elements does not exist if there are no group difficulties. It often functions as one of the group order of elements' divisors. If n is prime, the group $(Z_n\ ,\ \cdot)$ is cyclic. Else if n n is not prime, the group is generally acyclic. It's possible that certain components equal the entire row. Some elements' orders, however, are lower than the overall order. This circumstance was seen in both the G1 and G2 groups. We can thus derive this and build our pattern based on the outcomes of our solutions; the primary determinant of the existence of components equal to the total order is whether or not the group is cyclic.

3)

Thanks to group theory we can say that

$$x^{2003} \equiv x \ (mod\ 2003)$$

2003 group order Given that it is 2003, the group components' exponents are 2003 makes a 2003 mode loop.

We have this equation:

$$a \equiv b^{1621}$$

Here, we left *b* alone. If we can find the inverse of the 1621 in the 2003 mode, we can derive a general equality based on *a* and *b*.

$$b \equiv a^{1621^{-1}} \ (mod\ 2003)$$

We can obtain this value with the Extended Euclidean Algorithm.

$$2003 = 1 \cdot 1621 + 382$$
$$1621 = 4 \cdot 382 + 93$$
$$382 = 4 \cdot 93 + 10$$
$$93 = 9 \cdot 10 + 3$$
$$10 = 3 \cdot 3 + 1$$
$$3 = 3 \cdot 1 + 0$$

When the remainder becomes 0, the algorithm ends. Now we will find the inverse of the value 1621 by writing all the equations according to 1.

$$1 = 10 - 3 \cdot 3$$
$$1 = 10 - 3 \cdot (93 - 9 \cdot 10)$$
$$1 = 28 \cdot 10 - 3 \cdot 93$$
$$1 = 28 \cdot (382 - 4 \cdot 93) - 3 \cdot 93$$
$$1 = 488 \cdot 382 - 115 \cdot 1621$$
$$1 = 488 \cdot (2003 - 1 \cdot 1621) - 115 \cdot 1621$$
$$1 = 488 \cdot 2003 - 603 \cdot 1621$$

If we calculate the -603 value according to mod 2003, we get the value 1400.

$$b \equiv a^{1400} \ (mod\ 2003)$$

Using this equation, we can calculate the ***b*** value for any ***a*** element in the group.