Assignment 6.

Problem 1 (4.3)
Let $a, b \in \mathbb{Z}$ with $a \geq b \geq 0$, let $d := \gcd(a, b)$ and assume $d > 0$
Suppose that on input a, b, Euclid's Algorithm performs λ division steps, and computes the remainder sequence $\{r_i\}_{i=0}^{\lambda+1}$ and the quotient sequence $\{q_i\}_{i=1}^{\lambda}$ (as in Theorem 4.1) Now suppose we run Euclid's algorithm on input $a/d, b/d$.

Show that on these inputs, the number of division steps performed is also λ, the remainder sequence is $\{r_i/d\}_{i=0}^{\lambda+1}$ and the quotinet sequence is $\{q_i\}_{i=1}^{\lambda}$.

We know $a \geq b \geq 0$ $d = \gcd(a, b)$ $d > 0$
Run Euclid Algorithm for a, b
Since $a \geq b$

- $a = b \times q_1 + r_0$ $\quad 0 \leq r_0 < b$
- $b = r_0 \times q_2 + r_1$ $\quad 0 \leq r_1 < r_0$
- $r_0 = r_1 \times q_3 + r_2$ $\quad 0 \leq r_2 < r_1$
- $\vdots$
-
-
- $r_{\lambda-3} = r_{\lambda-2} \times q_\lambda + 0 (r_{\lambda-1})$ $\quad 0 \leq r_{\lambda-1} < r_{\lambda-2}$

It performs λ steps, get $\{r_i\}_{i=0}^{\lambda-1}$ and $\{q_i\}_{i=1}^{\lambda}$
For a/d, b/d

$$a/d = b/d \times q_1 + \frac{r_0}{d} \qquad 0 \leq \frac{r_0}{d} < b$$

$$\frac{b}{d} = \frac{r_0}{d} \times q_2 + \frac{r_1}{d} \qquad 0 \leq \frac{r_1}{d} < \frac{r_0}{d}$$

$\vdots$

$$\frac{r_{\lambda-3}}{d} = \frac{r_{\lambda-2}}{d} \times q_\lambda + \frac{r_{\lambda-1}}{d} \qquad 0 \leq \frac{r_{\lambda-1}}{d} \leq \frac{r_{\lambda-2}}{d}$$

Since From Euclid Algorithm for a, b r_{z-1} is 0 so in Euclid Algorithm of $a/d, b/d$, $\frac{r_{z-1}}{d}$ is also 0

so In the Euclid Algorithm for $a/d, b/d$ there are also z steps and remainder, quotient sequence for r_i and q_i are $\{r_i\}_{i=0}^{z-1}$, $\{q_i\}_{i=1}^{z}$

Problem 2 (4.9)
Assume notation as in Theorem 4.3 Show that

1. For all $i = 2, \dots, z$ we have $|t_i| < |t_{i+1}|$ and $r_{i-1}|t_i| < a$, and that for all $i = 3, \dots, z$, we have $|s_i| < |s_{i+1}|$ and $r_{i-1}|s_i| < b$;

2. $s_i t_i \leq 0$ for $i = 0, \dots, z+1$;

3. if $d = \gcd(a,b) > 0$, then $|s_{z+1}| = b/d$ and $|t_{z+1}| = a/d$

1. From $i = 0, \dots, z$ $t_i t_{i+1} \leq 0$ since t_i and t_{i+1} have different sign, and $|t_i| \leq |t_{i+1}|$. for $i = 1, \dots, z+1$ $r_{i-1}|t_i| \leq a$ since r_i is the remainders so this is true from the values of $i = 2, \dots, z$
so For $i = 1, \dots, z$, $|s_i| \leq |s_{i+1}|$ and $s_i s_{i+1} \leq 0$ (different sign), $r_{i-1}|s_i| \leq b$. for $i = 1, \dots, z$ $|s_i| < |s_{i+1}|$ and $r_{i-1}|s_i| < b$ when $i = 3, \dots, z$

2. From theorem 4.3 for $i = 0, \dots, z+1$ $\gcd(s_i, t_i) = 1$ they are relatively prime.
Let's say s_i and t_i are two values for $i = 0$. so $s_0 = 1$, $t_0 = 0$
$\gcd(1, 0) = 1$ as from the theorem

Similarly for $i = 1$, $s_1 = 0$ $t_1 = 1$ $s_i t_i = 0$ we can say $s_i t_i \leq 0$ for all values from $i = 0, \dots, z+1$ since $s_i t_i \leq 0$ for $i = 0, \dots, z+1$
if $s_i t_i \neq 0$ then it would be negative since they would have different sign

3. $\gcd(a,b) > 0$

we know $|S_i| \le b$ for $i = 1 \sim \lambda+1$. when $i = \lambda+1$, $|S_{\lambda+1}| \le b$

$\Rightarrow |S_{\lambda+1}| = b/d$ since $S_{\lambda+1}$ is the last S value, d is some greater than $0 = \gcd(a,b)$

For $i = 1 \sim \lambda+1$ $|t_{\lambda+1}| \le a$, for $d = \gcd(a,b) > 0$. $|t_{\lambda+1}| \le a/d$

so if $\gcd(a,b) > 0$, then $|S_{\lambda+1}| = b/d$ and $|t_{\lambda+1}|$ a/d.

Problem 3. (4.13)

In this exercise, you are to make the result of Theorem 2.17 effective. Suppose that we are given a positive integer n, two elements $\alpha, \beta \in \mathbb{Z}_n^*$, and integer l and m, such that $\alpha^l = \beta^m$ and $\gcd(l,m) = 1$. Show how to compute $r \in \mathbb{Z}_n^*$ such that $\alpha = r^m$ in time $O(\text{len}(l) \cdot \text{len}(m) + (\text{len}(l) + \text{len}(m)) \cdot \text{len}(n^2))$

From theorem 2.17 $\alpha^l = \beta^m \in (\mathbb{Z}_n^*)^m$ since $\gcd(l,m) = 1$. Since $\alpha = r^m$

$r^{-m \times m \bmod l} = r^1 = \alpha^{-m \bmod l}$ From fast Euclidean (Extended)

the run time of finding inverse of $m \bmod l$ is $\text{len}(m) \cdot \text{len}(l)$

then to use fast squaring Algorithm from chapter 3

α's power runtime is $\text{len}(l) \times \text{len}(n)^2$ so the total run time is

$O(\text{len}(l)\,\text{len}(m) + (\text{len}(l) + \text{len}(m)) \times \text{len}(n^2))$.