CSE 311 Section 5

Number Theory

Administrivia

Announcements & Reminders

- HW3
  - If you think something was graded incorrectly, submit a regrade request!
- HW4 was due yesterday
  - Use late days if you need them!
- HW5
  - Due next week on Oct 30th at 11:59PM on Gradescope

Greatest Common Divisor

Some Definitions

- Greatest Common Divisor (GCD):
  - The Greatest Common Divisor of a and b ($\gcd(a, b)$) is the largest integer c such that $c|a$ and $c|b$
- Multiplicative Inverse:
  - The multiplicative inverse of $a \pmod{n}$ is an integer b such that $ab \equiv 1 \pmod{n}$

Problem 1 – Warm-Up

a) Calculate gcd(100, 50).

a) Calculate gcd(17, 31)

a) Find the multiplicative inverse of 6 (mod 7).

a) Does 49 have a multiplicative inverse (mod 7)?

Try this problem with the people around you, and then we'll go over it together!

Problem 1 – Warm-Up

a) Calculate gcd(100, 50).

a) Calculate gcd(17, 31)

a) Find the multiplicative inverse of 6 (mod 7).

a) Does 49 have a multiplicative inverse (mod 7)?

Problem 1 – Warm-Up

a) Calculate gcd(100, 50).

50

a) Calculate gcd(17, 31)

a) Find the multiplicative inverse of 6 (mod 7).

a) Does 49 have a multiplicative inverse (mod 7)?

Problem 1 – Warm-Up

a) Calculate gcd(100, 50).

50

a) Calculate gcd(17, 31)

1

a) Find the multiplicative inverse of 6 (mod 7).

a) Does 49 have a multiplicative inverse (mod 7)?

Problem 1 – Warm-Up

a) Calculate gcd(100, 50).

50

a) Calculate gcd(17, 31)

1

a) Find the multiplicative inverse of 6 (mod 7).

6

a) Does 49 have a multiplicative inverse (mod 7)?

Problem 1 – Warm-Up

a) Calculate gcd(100, 50).

50

a) Calculate gcd(17, 31)

1

a) Find the multiplicative inverse of 6 (mod 7).

6

a) Does 49 have a multiplicative inverse (mod 7)?

It does not. Intuitively, this is because 49x for any x is going to be 0 mod 7, which means it can never be 1.

Extended Euclidean Algorithm

Finding GCD

GCD Facts:

If a and b are positive integers, then:

$$\gcd(a, b) = \gcd(b, a\%b)$$

$$\gcd(a, 0) = a$$

```
public int GCD(int m, int n){
    if(m<n){
        int temp = m;
        m=n;
        n=temp;
    }
    while(n != 0) {
        int rem = m % n;
        m=n;
        n=temp;
    }
    return m;
}
```

$$\gcd(a, b) = \gcd(b, a\%b)$$

Euclid's Algorithm

gcd(660,126)

$$\gcd(a, b) = \gcd(b, a\%b)$$

Euclid's Algorithm

gcd(660,126) = gcd(126, 660 % 126) = gcd(126, 30)

$$\gcd(a, b) = \gcd(b, a\%b)$$

Euclid's Algorithm

gcd(660,126) = gcd(126, 660 % 126) = gcd(126, 30)

= gcd(30, 126 % 30) = gcd(30, 6)

$$\gcd(a, b) = \gcd(b, a\%b)$$

Euclid's Algorithm

gcd(660,126) = gcd(126, 660 % 126) = gcd(126, 30)

= gcd(30, 126 % 30) = gcd(30, 6)

= gcd(6, 30 % 6) = gcd(6, 0)

$$\gcd(a, b) = \gcd(b, a\%b)$$

Euclid's Algorithm

gcd(660,126) = gcd(126, 660 % 126) = gcd(126, 30)
= gcd(30, 126 % 30) = gcd(30, 6)
= gcd(6, 30 % 6) = gcd(6, 0)
= 6

$$\gcd(a, b) = \gcd(b, a\%b)$$

Euclid's Algorithm

gcd(660,126) = gcd(126, 660 % 126) = gcd(126, 30)
= gcd(30, 126 % 30) = gcd(30, 6)
= gcd(6, 30 % 6) = gcd(6, 0)
= 6

Tableau form

$$660 = 5 \cdot 126 + 30$$
$$126 = 4 \cdot 30 + 6$$
$$30 = 5 \cdot 6 + 0$$

Bézout's Theorem

- Bézout's Theorem:
  - If a and b are positive integers, then there exist integers s and t such that

$$\gcd(a, b) = sa + tb$$

- We're not going to prove this theorem in section though, because it's hard and ugly

Extended Euclidean Algorithm

Bézout's Theorem tells us that $\gcd(a, b) = sa + tb$.

To find the s, t we can use the Extended Euclidean Algorithm.

- Step 1: compute $\gcd(a, b)$; keep tableau information
- Step 2: solve all equations for the remainder
- Step 3: substitute backward

Extended Euclidean Algorithm

gcd(35,27)

- **Compute $gcd(a, b)$; keep tableau information**
- Solve all equations for the remainder
- Substitute backward

Extended Euclidean Algorithm

gcd(35,27) = gcd(27, 35%27) = gcd(27,8)

- **Compute $gcd(a, b)$; keep tableau information**
- Solve all equations for the remainder
- Substitute backward

Extended Euclidean Algorithm

gcd(35,27) = gcd(27, 35%27) = gcd(27,8)
= gcd(8, 27%8) = gcd(8, 3)

- **Compute $gcd(a, b)$; keep tableau information**
- Solve all equations for the remainder
- Substitute backward

Extended Euclidean Algorithm

gcd(35,27) = gcd(27, 35%27) = gcd(27,8)
= gcd(8, 27%8) = gcd(8, 3)
= gcd(3, 8%3) = gcd(3, 2)

- **Compute $gcd(a, b)$; keep tableau information**
- Solve all equations for the remainder
- Substitute backward

Extended Euclidean Algorithm

gcd(35,27) = gcd(27, 35%27) = gcd(27,8)
= gcd(8, 27%8) = gcd(8, 3)
= gcd(3, 8%3) = gcd(3, 2)
= gcd(2, 3%2) = gcd(2,1)

- **Compute $gcd(a, b)$; keep tableau information**
- Solve all equations for the remainder
- Substitute backward

Extended Euclidean Algorithm

gcd(35,27) = gcd(27, 35%27) = gcd(27,8)
= gcd(8, 27%8) = gcd(8, 3)
= gcd(3, 8%3) = gcd(3, 2)
= gcd(2, 3%2) = gcd(2,1)
= gcd(1, 2%1) = gcd(1,0)

- **Compute $gcd(a, b)$; keep tableau information**
- Solve all equations for the remainder
- Substitute backward

Extended Euclidean Algorithm

- **Compute $gcd(a, b)$; keep tableau information**
- Solve all equations for the remainder
- Substitute backward

gcd(35,27)	= gcd(27, 35%27)	= gcd(27,8)
	= gcd(8, 27%8)	= gcd(8, 3)
	= gcd(3, 8%3)	= gcd(3, 2)
	= gcd(2, 3%2)	= gcd(2,1)
	= gcd(1, 2%1)	= gcd(1,0)

$$
\begin{aligned}
35 &= 1 \cdot 27 + 8 \\
27 &= 3 \cdot 8 + 3 \\
8 &= 2 \cdot 3 + 2 \\
3 &= 1 \cdot 2 + 1
\end{aligned}
$$

Extended Euclidean Algorithm

- Compute $gcd(a, b)$; keep tableau information
- **Solve all equations for the remainder**
- Substitute backward

$$
\begin{aligned}
35 &= 1 \cdot 27 + 8 \\
27 &= 3 \cdot 8 + 3 \\
8 &= 2 \cdot 3 + 2 \\
3 &= 1 \cdot 2 + 1
\end{aligned}
$$

Extended Euclidean Algorithm

- Compute $gcd(a, b)$; keep tableau information
- **Solve all equations for the remainder**
- Substitute backward

$$35 = 1 \cdot 27 + 8$$
$$27 = 3 \cdot 8 + 3$$
$$8 = 2 \cdot 3 + 2$$
$$3 = 1 \cdot 2 + 1$$

$$8 = 35 - 1 \cdot 27$$

Extended Euclidean Algorithm

- Compute $gcd(a, b)$; keep tableau information
- **Solve all equations for the remainder**
- Substitute backward

$$35 = 1 \cdot 27 + 8$$
$$27 = 3 \cdot 8 + 3$$
$$8 = 2 \cdot 3 + 2$$
$$3 = 1 \cdot 2 + 1$$

$$8 = 35 - 1 \cdot 27$$
$$3 = 27 - 3 \cdot 8$$

Extended Euclidean Algorithm

- Compute $gcd(a, b)$; keep tableau information
- **Solve all equations for the remainder**
- Substitute backward

$$
\begin{aligned}
35 &= 1 \cdot 27 + 8 \\
27 &= 3 \cdot 8 + 3 \\
8 &= 2 \cdot 3 + 2 \\
3 &= 1 \cdot 2 + 1
\end{aligned}
$$

$$
\begin{aligned}
8 &= 35 - 1 \cdot 27 \\
3 &= 27 - 3 \cdot 8 \\
2 &= 8 - 2 \cdot 3
\end{aligned}
$$

Extended Euclidean Algorithm

- Compute $gcd(a, b)$; keep tableau information
- **Solve all equations for the remainder**
- Substitute backward

$$
\begin{aligned}
35 &= 1 \cdot 27 + 8 \\
27 &= 3 \cdot 8 + 3 \\
8 &= 2 \cdot 3 + 2 \\
3 &= 1 \cdot 2 + 1
\end{aligned}
$$

$$
\begin{aligned}
8 &= 35 - 1 \cdot 27 \\
3 &= 27 - 3 \cdot 8 \\
2 &= 8 - 2 \cdot 3 \\
1 &= 3 - 1 \cdot 2
\end{aligned}
$$

Extended Euclidean Algorithm

```
8 = 35 - 1·27
3 = 27 - 3·8
2 = 8  - 2·3
1 = 3  - 1·2
```

- Compute $gcd(a, b)$; keep tableau information
- Solve all equations for the remainder
- **Substitute backward**

Extended Euclidean Algorithm

$$8 = 35 - 1 \cdot 27$$
$$3 = 27 - 3 \cdot 8$$
$$2 = 8 - 2 \cdot 3$$
$$1 = 3 - 1 \cdot 2$$

$$1 = 3 - 1 \cdot 2$$

- Compute $gcd(a, b)$; keep tableau information
- Solve all equations for the remainder
- **Substitute backward**

Extended Euclidean Algorithm

- Compute $gcd(a, b)$; keep tableau information
- Solve all equations for the remainder
- **Substitute backward**

```
8 = 35 - 1·27
3 = 27 - 3·8
2 = 8  - 2·3
1 = 3  - 1·2
```

```
1 = 3 - 1·2
  = 3 - 1·(8 - 2·3)
```

Extended Euclidean Algorithm

- Compute $gcd(a, b)$; keep tableau information
- Solve all equations for the remainder
- **Substitute backward**

```
8 = 35 - 1·27
3 = 27 - 3·8
2 = 8  - 2·3
1 = 3  - 1·2
```

```
1 = 3 - 1·2
  = 3 - 1·(8 - 2·3)
  = -1·8 + 3·3
```

Extended Euclidean Algorithm

- Compute $gcd(a, b)$; keep tableau information
- Solve all equations for the remainder
- **Substitute backward**

```
8 = 35 - 1·27
3 = 27 - 3·8
2 = 8  - 2·3
1 = 3  - 1·2
```

```
1 = 3 - 1·2
  = 3 - 1·(8 - 2·3)
  = -1·8 + 3·3
  = -1·8 + 3(27 - 3·8)
```

Extended Euclidean Algorithm

- Compute $gcd(a, b)$; keep tableau information
- Solve all equations for the remainder
- **Substitute backward**

```
8 = 35 - 1·27
3 = 27 - 3·8
2 = 8  - 2·3
1 = 3  - 1·2
```

```
1 = 3 - 1·2
  = 3 - 1·(8 - 2·3)
  = -1·8 + 3·3
  = -1·8 + 3(27 - 3·8)
  = 3·27 - 10·8
```

Extended Euclidean Algorithm

- Compute $gcd(a, b)$; keep tableau information
- Solve all equations for the remainder
- **Substitute backward**

```
8 = 35 - 1·27
3 = 27 - 3·8
2 = 8  - 2·3
1 = 3  - 1·2
```

```
1 = 3 - 1·2
  = 3 - 1·(8 - 2·3)
  = -1·8 + 3·3
  = -1·8 + 3(27 - 3·8)
  = 3·27 - 10·8
  = 3·27 - 10(35 - 1·27)
```

Extended Euclidean Algorithm

- Compute $gcd(a, b)$; keep tableau information
- Solve all equations for the remainder
- **Substitute backward**

```
8 = 35 - 1·27
3 = 27 - 3·8
2 = 8  - 2·3
1 = 3  - 1·2
```

```
1 = 3 - 1·2
  = 3 - 1·(8 - 2·3)
  = -1·8 + 3·3
  = -1·8 + 3(27 - 3·8)
  = 3·27 - 10·8
  = 3·27 - 10(35 - 1·27)
  = 13·27 - 10·35
```

Extended Euclidean Algorithm

- Compute $gcd(a, b)$; keep tableau information
- Solve all equations for the remainder
- **Substitute backward**

$$
\begin{aligned}
8 &= 35 - 1 \cdot 27 \\
3 &= 27 - 3 \cdot 8 \\
2 &= 8 - 2 \cdot 3 \\
1 &= 3 - 1 \cdot 2
\end{aligned}
$$

When substituting back, you keep the larger of m, n and the number you just substituted.

Don't simplify further! (or you'll lose the form you need)

$$
\begin{aligned}
1 &= 3 - 1 \cdot 2 \\
&= 3 - 1 \cdot (8 - 2 \cdot 3) \\
&= -1 \cdot 8 + 3 \cdot 3 \\
&= -1 \cdot 8 + 3(27 - 3 \cdot 8) \\
&= 3 \cdot 27 - 10 \cdot 8 \\
&= 3 \cdot 27 - 10(35 - 1 \cdot 27) \\
&= 13 \cdot 27 - 10 \cdot 35
\end{aligned}
$$

Problem 2 – Extended Euclidean Algorithm

a) Find the multiplicative inverse y of $7 \bmod 33$. That is, find y such that $7y \equiv 1 \ (mod \ 33)$. You should use the extended Euclidean Algorithm. Your answer should be in the range $0 \leq y < 33$.

b) Now, solve $7z \equiv 2 \ (\mathrm{mod}\ 33)$ for all of its integer solutions z.

Try this problem with the people around you, and then we'll go over it together!

Problem 2 – Extended Euclidean Algorithm

a) Find the multiplicative inverse y of $7 \bmod 33$. That is, find y such that $7y \equiv 1 \ (mod \ 33)$. You should use the extended Euclidean Algorithm. Your answer should be in the range $0 \leq y < 33$.

Problem 2 – Extended Euclidean Algorithm

a) Find the multiplicative inverse y of $7 \bmod 33$. That is, find y such that $7y \equiv 1 \ (mod\ 33)$. You should use the extended Euclidean Algorithm. Your answer should be in the range $0 \leq y < 33$.

First, we find the gcd:

```
gcd(33,7)   = gcd(7,5)      33 = 4 • 7 + 5
            = gcd(5,2)      7  = 1 • 5 + 2
            = gcd(2,1)      5  = 2 • 2 + 1
            = gcd(1,0)      2  = 2 • 1 + 0
```

Problem 2 – Extended Euclidean Algorithm

a) Find the multiplicative inverse y of $7 \bmod 33$. That is, find y such that $7y \equiv 1 \ (mod\ 33)$. You should use the extended Euclidean Algorithm. Your answer should be in the range $0 \leq y < 33$.

First, we find the gcd:

```
gcd(33,7)   = gcd(7,5)      33 = 4 • 7 + 5
            = gcd(5,2)      7  = 1 • 5 + 2
            = gcd(2,1)      5  = 2 • 2 + 1
            = gcd(1,0)      2  = 2 • 1 + 0
```

Next, we re-arrange the equations by solving for the remainder:

```
1 = 5 − 2 • 2
2 = 7 − 1 • 5
5 = 33 − 4 • 7
```

Problem 2 – Extended Euclidean Algorithm

a) Find the multiplicative inverse y of $7 \bmod 33$. That is, find y such that $7y \equiv 1 \ (mod\ 33)$. You should use the extended Euclidean Algorithm. Your answer should be in the range $0 \leq y < 33$.

First, we find the gcd:

```
gcd(33,7)   = gcd(7,5)      33 = 4 • 7 + 5
            = gcd(5,2)      7  = 1 • 5 + 2
            = gcd(2,1)      5  = 2 • 2 + 1
            = gcd(1,0)      2  = 2 • 1 + 0
```

Next, we re-arrange the equations by solving for the remainder:

```
1 = 5 − 2 • 2
2 = 7 − 1 • 5
5 = 33 − 4 • 7
```

Now, we backward substitute into the boxed numbers using the equations:

$$
\begin{aligned}
1 &= 5 - 2 \cdot 2 \\
&= 5 - 2 \cdot (7 - 1 \cdot 5) \\
&= 3 \cdot 5 - 2 \cdot 7 \\
&= 3 \cdot (33 - 4 \cdot 7) - 2 \cdot 7 \\
&= 3 \cdot 33 + -14 \cdot 7
\end{aligned}
$$

Problem 2 – Extended Euclidean Algorithm

a) Find the multiplicative inverse y of 7 mod 33. That is, find y such that $7y \equiv 1 \ (mod\ 33)$. You should use the extended Euclidean Algorithm. Your answer should be in the range $0 \leq y < 33$.

First, we find the gcd:

```
gcd(33,7)   = gcd(7,5)
            = gcd(5,2)
            = gcd(2,1)
            = gcd(1,0)
```

```
33 = 4 • 7 + 5
7  = 1 • 5 + 2
5  = 2 • 2 + 1
2  = 2 • 1 + 0
```

Next, we re-arrange the equations by solving for the remainder:

```
1 = 5 − 2 • 2
2 = 7 − 1 • 5
5 = 33 − 4 • 7
```

Now, we backward substitute into the boxed numbers using the equations:

```
1  = 5 − 2 • 2
   = 5 −  2 •  (7 − 1 • 5)
   = 3 • 5 − 2 • 7
   = 3 • (33 − 4 • 7) − 2 •  7
   = 3 • 33 + −14 • 7
```

So, 1 = 33 • 3 + 7 • −14. Thus, 33 − 14 = 19 is the multiplicative inverse of 7 mod 33

Problem 2 – Extended Euclidean Algorithm

b) Now, solve $7z \equiv 2 \pmod{33}$ for all of its integer solutions z.

Problem 2 – Extended Euclidean Algorithm

b) Now, solve $7z \equiv 2 \pmod{33}$ for all of its integer solutions z.

If $7y \equiv 1 \pmod{33}$, then $2 \cdot 7y \equiv 2 \pmod{33}$.

Problem 2 – Extended Euclidean Algorithm

b) Now, solve $7z \equiv 2 \pmod{33}$ for all of its integer solutions z.

If $7y \equiv 1 \pmod{33}$, then $2 \cdot 7y \equiv 2 \pmod{33}$.

So, $z \equiv 2 \cdot 19 \pmod{33} \equiv 5 \pmod{33}$. This means that the set of solutions is $\{5 + 33k \mid k \in Z\}$

Proof by Contradiction

5b: argue code is valid using contradiction

Problem 5b - Proof by Contradiction

```
public boolean isPrime(int n) {
    int potentialDiv = 2;
    while (potentialDiv < n) {
        if (n % potentialDiv == 0)
            return false;
        potentialDiv++;
    }
    return true;
}
```

(Given)

Returns true if and only if *n* is prime (assume n > 0).

Problem 5b - Proof by Contradiction

```
public boolean isPrime(int n) {
    int potentialDiv = 2;
    while (potentialDiv <= Math.sqrt(n)) {
        if (n % potentialDiv == 0)
            return false;
        potentialDiv++;
    }
    return true;
}
```

(Modified)

(*Maybe*) Returns true if and only if *n* is prime (assume n > 0).

Problem 5b - Proof by Contradiction

“nontrivial divisor”: a factor that isn’t 1 or the number itself. Formally, a positive integer k being a “nontrivial divisor” of n means that $k|n$, $k \neq 1$ and $k \neq n$.

Claim: For every positive integer n, if n has a nontrivial divisor, then it has a nontrivial divisor at most $\sqrt{n}$

Prove this with contradiction!

Problem 5b - Proof by Contradiction

Claim: For every positive integer n, if n has a nontrivial divisor, then it has a nontrivial divisor at most √n

(proof by contradiction): Suppose, for the sake of contradiction, that there is an n such that n has a non-trivial divisor and all its nontrivial divisors are greater than $\sqrt{n}$.

Problem 5b - Proof by Contradiction

Claim: For every positive integer n, if n has a nontrivial divisor, then it has a nontrivial divisor at most $\sqrt{n}$

(proof by contradiction): Suppose, for the sake of contradiction, that there is an n such that n has a non-trivial divisor and all its nontrivial divisors are greater than $\sqrt{n}$.

Let k be a nontrivial divisor of n. Since k is a divisor, $n = kc$ for some integer c. Observe that c is also nontrivial, since if c were 1 or n then k would have to be n or 1.

Problem 5b - Proof by Contradiction

Claim: For every positive integer n, if n has a nontrivial divisor, then it has a nontrivial divisor at most √n

(proof by contradiction): Suppose, for the sake of contradiction, that there is an n such that n has a non-trivial divisor and all its nontrivial divisors are greater than $\sqrt{n}$.

Let k be a nontrivial divisor of n. Since k is a divisor, $n = kc$ for some integer c. Observe that c is also nontrivial, since if c were 1 or n then k would have to be n or 1.

Since both k and n are non-trivial divisors, we have that $k > \sqrt{n}$ and $c > \sqrt{n}$. Then $kc > \sqrt{n}\sqrt{n} = n$. But by assumption we have $kc = n$, so this is a contradiction. Thus we conclude our original claim—that if a positive integer n has a nontrivial divisor, then it has a nontrivial divisor at most $\sqrt{n}$—is true.

Problem 5b - Proof by Contradiction, v2

Claim: For every positive integer n, if n has a nontrivial divisor, then it has a nontrivial divisor at most √n

(alternative proof): Let k be a nontrivial divisor of n. Since k is a divisor, $n = kc$ for some integer c. Observe that c is also nontrivial, since if c were 1 or n then k would have to be n or 1.

Problem 5b - Proof by Contradiction, v2

Claim: For every positive integer n, if n has a nontrivial divisor, then it has a nontrivial divisor at most √n

(alternative proof): Let k be a nontrivial divisor of n. Since k is a divisor, $n = kc$ for some integer c. Observe that c is also nontrivial, since if c were 1 or n then k would have to be n or 1.

We now have two cases:

Case 1: $k \leq \sqrt{n}$

Problem 5b - Proof by Contradiction, v2

Claim: For every positive integer n, if n has a nontrivial divisor, then it has a nontrivial divisor at most √n

(alternative proof): Let k be a nontrivial divisor of n. Since k is a divisor, $n = kc$ for some integer c. Observe that c is also nontrivial, since if c were 1 or n then k would have to be n or 1.

We now have two cases:

Case 1: $k \leq \sqrt{n}$
If $k \leq \sqrt{n}$, then we're done because k is the desired nontrivial divisor.

Problem 5b - Proof by Contradiction, v2

Claim: For every positive integer n, if n has a nontrivial divisor, then it has a nontrivial divisor at most √n

(alternative proof): Let k be a nontrivial divisor of n. Since k is a divisor, $n = kc$ for some integer c. Observe that c is also nontrivial, since if c were 1 or n then k would have to be n or 1.

We now have two cases:

Case 1: $k \leq \sqrt{n}$
If $k \leq \sqrt{n}$, then we're done because k is the desired nontrivial divisor.

Case 2: $k > \sqrt{n}$

Problem 5b - Proof by Contradiction, v2

Claim: For every positive integer n, if n has a nontrivial divisor, then it has a nontrivial divisor at most √n

(alternative proof): Let k be a nontrivial divisor of n. Since k is a divisor, $n = kc$ for some integer c. Observe that c is also nontrivial, since if c were 1 or n then k would have to be n or 1.

We now have two cases:

Case 1: $k \leq \sqrt{n}$
If $k \leq \sqrt{n}$, then we're done because k is the desired nontrivial divisor.

Case 2: $k > \sqrt{n}$
If $k > \sqrt{n}$, then multiplying both sides by c we get $ck > c\sqrt{n}$. But $ck = n$ so $n > c\sqrt{n}$. Finally, dividing both sides by $\sqrt{n}$ gives $\sqrt{n} > c$, so c is the desired nontrivial factor.

Problem 5b - Proof by Contradiction, v2

Claim: For every positive integer n, if n has a nontrivial divisor, then it has a nontrivial divisor at most √n

(alternative proof): Let k be a nontrivial divisor of n. Since k is a divisor, $n = kc$ for some integer c. Observe that c is also nontrivial, since if c were 1 or n then k would have to be n or 1.

We now have two cases:

Case 1: $k \leq \sqrt{n}$
If $k \leq \sqrt{n}$, then we're done because k is the desired nontrivial divisor.

Case 2: $k > \sqrt{n}$
If $k > \sqrt{n}$, then multiplying both sides by c we get $ck > c\sqrt{n}$. But $ck = n$ so $n > c\sqrt{n}$. Finally, dividing both sides by $\sqrt{n}$ gives $\sqrt{n} > c$, so c is the desired nontrivial factor.

In both cases we find a nontrivial divisor at most $\sqrt{n}$, as required.

Number Theory

Bonus! :D

Some Definitions

- Divides:
  - For $a, b \in \mathbb{Z}$: $a \mid b$ iff $\exists (k \in \mathbb{Z})\ b = ka$
  - For integers a and b, we say a divides b if and only if there exists an integer k such that $b = ka$
- Congruence Modulo:
  - For $a, b \in \mathbb{Z}, m \in \mathbb{Z}^+$: $a \equiv b \pmod{m}$ iff $m \mid (b - a)$
  - For integers a and b and positive integer m, we say a is congruent to b modulo m if and only if m divides $b - a$

Problem 4 – Modular Arithmetic

a) Prove that if $a \mid b$ and $b \mid a$, where a and b are integers, then $a = b$ or $a = -b$.

b) Prove that if $n \mid m$, where n and m are integers greater than 1, and if $a \equiv b \pmod{m}$, where a and b are integers, then $a \equiv b \pmod{n}$.

Lets walk through part (a) together.

Problem 4 – Modular Arithmetic

a) Prove that if $a \mid b$ and $b \mid a$, where a and b are integers, then $a = b$ or $a = -b$.

Suppose that $a \mid b$ and $b \mid a$, where a, b are integers.

…

Therefore, it follows that $a = -b$ or $a = b$.

Start with your proof skeleton!

Problem 4 – Modular Arithmetic

a) Prove that if $a \mid b$ and $b \mid a$, where a and b are integers, then $a = b$ or $a = -b$.

Suppose that $a \mid b$ and $b \mid a$, where a, b are integers.

By the definition of divides, we have $a \neq 0$, $b \neq 0$ and $b = ka$, $a = jb$ for some integers k, j.

…

Therefore, it follows that $a = -b$ or $a = b$.

Problem 4 – Modular Arithmetic

a) Prove that if $a \mid b$ and $b \mid a$, where a and b are integers, then $a = b$ or $a = -b$.

Suppose that $a \mid b$ and $b \mid a$, where a, b are integers.

By the definition of divides, we have $a \neq 0, b \neq 0$ and $b = ka, a = jb$ for some integers k, j.
Combining these equations, we see that $a = j(ka)$.
…

Therefore, it follows that $a = -b$ or $a = b$.

Problem 4 – Modular Arithmetic

a) Prove that if $a \mid b$ and $b \mid a$, where a and b are integers, then $a = b$ or $a = -b$.

Suppose that $a \mid b$ and $b \mid a$, where a, b are integers.

By the definition of divides, we have $a \neq 0$, $b \neq 0$ and $b = ka$, $a = jb$ for some integers k, j.
Combining these equations, we see that $a = j(ka)$.
Then, dividing both sides by a, we get $1 = jk$. So, $\frac{1}{j} = k$.

…

Therefore, it follows that $a = -b$ or $a = b$.

Problem 4 – Modular Arithmetic

a) Prove that if $a \mid b$ and $b \mid a$, where a and b are integers, then $a = b$ or $a = -b$.

Suppose that $a \mid b$ and $b \mid a$, where a, b are integers.

By the definition of divides, we have $a \neq 0$, $b \neq 0$ and $b = ka$, $a = jb$ for some integers k, j.
Combining these equations, we see that $a = j(ka)$.
Then, dividing both sides by a, we get $1 = jk$. So, $\frac{1}{j} = k$.
Note that j and k are integers, which is only possible if $j, k \in \{1, -1\}$.

Therefore, it follows that $a = -b$ or $a = b$.

Problem 4 – Modular Arithmetic

a) Prove that if $a \mid b$ and $b \mid a$, where a and b are integers, then $a = b$ or $a = -b$.

b) Prove that if $n \mid m$, where n and m are integers greater than 1, and if $a \equiv b \pmod{m}$, where a and b are integers, then $a \equiv b \pmod{n}$.

Now try part (b) with the people around you, and then we'll go over it together!

Problem 4 – Modular Arithmetic

b) Prove that if $n \mid m$, where n and m are integers greater than 1, and if $a \equiv b \pmod{m}$, where a and b are integers, then $a \equiv b \pmod{n}$.

Problem 4 – Modular Arithmetic

b) Prove that if $n \mid m$, where n and m are integers greater than 1, and if $a \equiv b \pmod{m}$, where a and b are integers, then $a \equiv b \pmod{n}$.

Let n, m, a, b be integers. Suppose $n \mid m$ with $n, m > 1$, and $a \equiv b \pmod{m}$.

…

Therefore, we have $a \equiv b \pmod{n}$.

Problem 4 – Modular Arithmetic

b) Prove that if $n \mid m$, where n and m are integers greater than 1, and if $a \equiv b \pmod{m}$, where a and b are integers, then $a \equiv b \pmod{n}$.

Let n, m, a, b be integers. Suppose $n \mid m$ with $n, m > 1$, and $a \equiv b \pmod{m}$.

…

… we have $n \mid (b - a)$.

Therefore, by definition of congruence, we have $a \equiv b \pmod{n}$.

Problem 4 – Modular Arithmetic

b) Prove that if $n \mid m$, where n and m are integers greater than 1, and if $a \equiv b \pmod{m}$, where a and b are integers, then $a \equiv b \pmod{n}$.

Let n, m, a, b be integers. Suppose $n \mid m$ with $n, m > 1$, and $a \equiv b \pmod{m}$.

NOTE: we don't know what C will look like yet, just that there is SOME integer here!

…

… we have $b - a = nC$.

Because C is an integer, by definition of divides, we have $n \mid (b - a)$.

Therefore, by definition of congruence, we have $a \equiv b \pmod{n}$.

Problem 4 – Modular Arithmetic

b) Prove that if $n \mid m$, where n and m are integers greater than 1, and if $a \equiv b \pmod{m}$, where a and b are integers, then $a \equiv b \pmod{n}$.

Let n, m, a, b be integers. Suppose $n \mid m$ with $n, m > 1$, and $a \equiv b \pmod{m}$.

By definition of divides, we have $m = kn$ for some $k \in \mathbb{Z}$.

…

… we have $b - a = nC$.

Because C is an integer, by definition of divides, we have $n \mid (b - a)$.

Therefore, by definition of congruence, we have $a \equiv b \pmod{n}$.

Problem 4 – Modular Arithmetic

b) Prove that if $n \mid m$, where n and m are integers greater than 1, and if $a \equiv b \pmod{m}$, where a and b are integers, then $a \equiv b \pmod{n}$.

Let n, m, a, b be integers. Suppose $n \mid m$ with $n, m > 1$, and $a \equiv b \pmod{m}$.

By definition of divides, we have $m = kn$ for some $k \in \mathbb{Z}$.
By definition of congruence, we have $m \mid a - b$, which means that $a - b = mj$ for some $j \in \mathbb{Z}$.
…
… we have $b - a = nC$.
Because C is an integer, by definition of divides, we have $n \mid (b - a)$.

Therefore, by definition of congruence, we have $a \equiv b \pmod{n}$.

Problem 4 – Modular Arithmetic

b) Prove that if $n \mid m$, where n and m are integers greater than 1, and if $a \equiv b \pmod{m}$, where a and b are integers, then $a \equiv b \pmod{n}$.

Let n, m, a, b be integers. Suppose $n \mid m$ with $n, m > 1$, and $a \equiv b \pmod{m}$.

By definition of divides, we have $m = kn$ for some $k \in \mathbb{Z}$.
By definition of congruence, we have $m \mid a - b$, which means that $a - b = mj$ for some $j \in \mathbb{Z}$.
Combining the two equations, we see that $a - b = (knj) = n(kj)$.
… we have $b - a = nC$.
Because C is an integer, by definition of divides, we have $n \mid (b - a)$.

Therefore, by definition of congruence, we have $a \equiv b \pmod{n}$.

Problem 4 – Modular Arithmetic

b) Prove that if $n \mid m$, where n and m are integers greater than 1, and if $a \equiv b \pmod{m}$, where a and b are integers, then $a \equiv b \pmod{n}$.

Let n, m, a, b be integers. Suppose $n \mid m$ with $n, m > 1$, and $a \equiv b \pmod{m}$.

By definition of divides, we have $m = kn$ for some $k \in \mathbb{Z}$.
By definition of congruence, we have $m \mid a - b$, which means that $a - b = mj$ for some $j \in \mathbb{Z}$.
Combining the two equations, we see that $a - b = (knj) = n(kj)$.
Equivalently, we have $b - a = n(-kj)$.
Because C is an integer, by definition of divides, we have $n \mid (b - a)$.

Therefore, by definition of congruence, we have $a \equiv b \pmod{n}$.

Problem 4 – Modular Arithmetic

b) Prove that if $n \mid m$, where n and m are integers greater than 1, and if $a \equiv b \pmod{m}$, where a and b are integers, then $a \equiv b \pmod{n}$.

Let n, m, a, b be integers. Suppose $n \mid m$ with $n, m > 1$, and $a \equiv b \pmod{m}$.

By definition of divides, we have $m = kn$ for some $k \in \mathbb{Z}$.
By definition of congruence, we have $m \mid a - b$, which means that $a - b = mj$ for some $j \in \mathbb{Z}$.
Combining the two equations, we see that $a - b = (knj) = n(kj)$.
Equivalently, we have $b - a = n(-kj)$.
Because $-kj$ is an integer, by definition of divides, we have $n \mid (b - a)$.

Therefore, by definition of congruence, we have $a \equiv b \pmod{n}$.

That's All, Folks!

Thanks for coming to section this week!
Any questions?