

added by me

not added by me

LEGEND

Greatest Common Divisor (GCD)

or "gcd"
$$\text{GCD}(A, B) = C$$

$$\text{gcd}(24, 16) = 8$$

$$\text{gcd}(15, 32) = 1$$

$$\text{gcd}(14, 27) = 1$$

$$\text{gcd}(15, 120) = 15$$

$$\forall A, B, C \in \mathbb{Z}$$

$$\exists C \in \mathbb{Z} \text{ s.t. } C \mid A \wedge C \mid B$$

$$\wedge \nexists D \in \mathbb{Z} \text{ s.t. } D \mid A \wedge D \mid B \wedge D > C$$

CODE ATTEMPT#1:

$$n = \max(a, b)$$

```
def gcd(a, b):  
    cur_gcd = 1;
```

```
    for i = 1 → min(a, b):  
        if i | a and i | b:  
            cur_gcd = i;
```

```
    return cur_gcd
```

$[O(\min(a, b))]$

$O(n)$ worst case

$$\gcd(a, b) = k = \gcd(a-b, b)$$

$$a = kc, \quad b = kd$$

$$\gcd(kc, kd) = k = \gcd(k(c-d), kd)$$

$$\gcd(a, b) = a = b \text{ WHEN } a \equiv b$$

↑
long equality
from JavaScript

CODE ATTEMPT #2:

```
def gcd(a, b)
```

```
    if a ≡ b return a; // or b
```

```
    else return a;
```

```
        if a > b return gcd(a-b, b);
```

```
        else return gcd(a, b-a);
```

$\gcd(a, b) \ a > b$ Without Loss of Generality (WLOG)

$$\gcd(a, b) = \gcd(a, r) \quad b = ak + r$$

CODE ATTEMPT #3

```
def gcd(a, b)
```

```
    if  $b \equiv 0$  return a;
```

```
    else  $r = a \% b$ 
```

↑

// mod symbol

```
    return gcd(b, r)
```

$\gcd(240, 46)$ // v3

$$240 = 5 \cdot 46 + 10$$

$$46 = 4 \cdot 10 + 6$$

$$10 = 1 \cdot 6 + 4$$

$$6 = 1 \cdot 4 + 2$$

$$4 = 2 \cdot 2 + 0$$

RETURN (2)

$$\forall a, b \in \mathbb{N}_{>0}$$

$$\exists x, y \in \mathbb{Z} \text{ st } x \cdot a + y \cdot b = \gcd(a, b)$$

$$S' = \{ x a + y b \mid x, y \in \mathbb{Z} \}$$

$$S = S' \cap \mathbb{N}_{>0}$$

S HAS AT LEAST ELEMENTS:

POSITIVE LINEAR COMBINATIONS

$$x=1, y=0 \rightarrow a$$

$$x=0, y=1 \rightarrow b$$

$$x=y=1 \rightarrow a+b$$

GOAL: SHOW THAT SMALLEST ELEMENT IN
 S IS $\gcd(a, b)$

LET d BE THE SMALLEST IN S

- 1) $d \mid a \wedge d \mid b \rightarrow d$ IS A DIVISOR OF a & b
- 2) d IS THE GREATEST DIVISOR OF a & b

SHOW $d \mid a$ $r \in [0, d) \cap \mathbb{N}$

$$d = ax + by$$

$$a = qd + r$$

$$d = x(qd + r) + by$$

$$d = xqd + xr + by$$

$$d - xqd = xr + by$$

$$d = \frac{xr + by}{1 - xq}$$

$$r = d - qd$$

$$r = d - q(ax + by)$$

$$r = d(1 - qx) - b(qy)$$

IF $r > 0$ SINCE $r < d$

ELEMENT $W.S$ THAT IS SMALLER THAN d

CONTRADICTION $\therefore r = 0$

$$\forall d' \in \mathbb{N}_{>0} \text{ st } d' \mid d \wedge d' \mid b$$

$$d' \mid ax + by$$

$$\therefore d' \mid d$$

$$d \geq d'$$

CODE ATTEMPT #4

def extended_gcd(a, b):

if $b \equiv 0$ return (a, 1, 0)

(g, s, t) = extended_gcd(b, a % b);

a, a % b $\rightarrow$ d, x, y

a, a - kb d = ax + by

$$d = xb + y(a - kb)$$

$$d = ax + b(x - ky)$$

$$x = t$$

$$y = (t - \lfloor b/a \rfloor t) = t(1 - \lfloor b/a \rfloor)$$

$$r = a \% b, \quad q = b \% r = b \% (a \% b)$$
$$(a, b) \rightarrow (b, r) \rightarrow (r, q)$$

$$a = bq + r$$

$$a \geq b + r > 2r$$

$$r \leq \frac{1}{2}a$$

HALVES EVERY 2 STEPS

COT3100C

Handwritten notes typed up and expanded

Instructor: Karthik Gajulapalli

Date: 10/01/2026

Scribe: Alexander Velazquez

Legend

Untouched original text (*#000000*)

Inserted subsection header (*#000000 bold*)

Added to original text (*#4EA72E*)

Modified or significantly augmented original text (*#0E2841*)

Greatest Common Divisor (GCD)

or "gcd"; *case insensitive*

$\text{GCD}(A, B) = C$

$$\text{gcd}(24, 16) = 8$$

$$\text{gcd}(15, 32) = 1$$

$$\text{gcd}(19, 27) = 1$$

$$\text{gcd}(15, 120) = 15$$

$\forall A, B \in \mathbb{N}$

$\exists C \in \mathbb{N} \text{ s.t. } \{ C \mid A \wedge C \mid B \}$

$\wedge \nexists D \in \mathbb{N} \text{ s.t. } \{ D \mid A \wedge D \mid B \wedge D > C \}$

CODE ATTEMPT #1:

```
def gcd(a, b)
```

```
    cur_gcd = 1;
```

```
    for i = 1 → min(a, b):
```

```
        if i | a ∧ i | b
```

```
            cur_gcd = i;
```

```
return cur_gcd;
```

Time complexity

$n = \max(a, b)$

$[O(\min(a, b))]$

$O(n)$ worst case

$\gcd(a, b) = k = \gcd(a - b, b)$

$a = kc, b = kd; c, d \in \mathbb{N}$

$\gcd(kc, kd) = k = \gcd(k(c-d), kd)$

$\gcd(a, b) = a = b$ WHEN $a \equiv b$

long "triple" equal sign from JavaScript; it will be used for conditions to distinguish from definitions and "setting" variables

CODE ATTEMPT #2:

```
def gcd(a, b)
```

```
    if a  $\equiv$  b return a; // OR b
```

```
    else
```

```
        if a > b return gcd(a - b, b);
```

```
        else return gcd(a, b - a);
```

$a > b$ Without Loss of Generality (WLoG)

$b = ak + r$

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

CODE ATTEMPT #3:

```
def gcd(a, b)
```

```
    if b  $\equiv$  0 return a;
```

```
    else
```

```
        r = a % b;
```

```
        return gcd(b, r);
```

$\text{gcd}(240, 46) // \text{V3}$

$$240 = 5 \cdot 46 + 10$$

$$46 = 4 \cdot 10 + 6$$

$$10 = 1 \cdot 6 + 4$$

$$6 = 1 \cdot 4 + 2$$

$$4 = 2 \cdot 2 + 0$$

RETURN 2 [RETURNING **a** BECAUSE **b** HAS REACHED **0**]

$$\forall a, b \in \mathbb{N}_{>0}$$

$$\exists x, y \in \mathbb{Z} \text{ s.t. } x \cdot a + y \cdot b = \text{gcd}(a, b)$$

$$S' = \{ xa + yb \mid x, y \in \mathbb{Z} \}$$

$$S = S' \cap \mathbb{N}_{>0}$$

S HAS AT **LEAST** ELEMENTS:

POSITIVE LINEAR COMBINATIONS

$$x = 1, y = 0 \rightarrow a$$

$$x = 0, y = 1 \rightarrow b$$

$$x = y = 1 \rightarrow a + b$$

GOAL: SHOW THAT SMALLEST ELEMENT IN S IS $\text{gcd}(a, b)$

LET **d** BE THE SMALLEST IN S

1) $d \mid a \wedge d \mid b \rightarrow d$ IS A DIVISOR OF **a** & **b**

2) **d** IS THE GREATEST DIVISOR OF **a** & **b**

SHOW $d \mid a$

$$r \in [0, d) \cap \mathbb{N}_{>0}$$

$$d = ax + by$$

$$q \in \mathbb{N}_{>0}$$

$$a = qd + r$$

$$d = x(qd + r) + by$$

$$d = xqd + xr + by$$

$$d - xqd = xr + by$$

$$d = \frac{xr + by}{1 - xq}$$

$$r = a - qd$$

$$r = a - q(ax + by)$$

$$r = a(1 - qx) - b(qy)$$

IF $r > 0$, SINCE $r < d$, IT WOULD BE AN ELEMENT IN S THAT IS SMALLER THAN d

THIS IS A CONTRADICTION WITH d BEING THE SMALLEST ELEMENT IN S , $\therefore r = 0 \therefore d \mid a$

$$\forall d' \in \mathbb{N}_{>0} \text{ s.t. } d' \mid a \wedge d' \mid b$$

$$d' \mid ax + by$$

$$d' \mid d$$

$$d \geq d' \text{ (CONTRADICTION)}$$

CODE ATTEMPT / SAMPLE #4

```
def extended_gcd(a, b)
```

```
    if b == 0 return (a, 1, 0);
```

```
    (g, s, t) = extended_gcd(b, a % b);
```

```
    // Apparently we stopped here to look at g, s, and t instead of finishing
```

```
    // This is just the same as V3 so it doesn't matter
```

```
    a, a % b → d, x, y
```

```
    d = ax + by
```

```
    a, a - kb
```

```
    d = xb + y(a - kb)
```

$$d = ay + b(x - ky)$$

$$x = t$$

$$y = (t - \lfloor b/a \rfloor t) = t(1 - \lfloor b/a \rfloor)$$

Time complexity

DIFFERENT **q** FROM BEFORE

$$r = a \% b, q = b \% r = b \% (a \% b)$$

$$(a, b) \rightarrow (b, r) \rightarrow (r, q)$$

$$a = bk + r$$

$$a \geq b + r > 2r$$

$$r \leq \frac{1}{2}a$$

HALVED EVERY 2 STEPS

$$O(2\log_2 n) \text{ OR } O(\log_2 n^2)$$