

Euclid's Algorithm

✔ **Video:** Greatest Common Divisor
10 min

✔ **Practice Quiz:** Greatest Common Divisor
3 questions

✔ **Video:** Euclid's Algorithm
15 min

✔ **Reading:** Greatest Common Divisor: Code
15 min

▶ **Video:** Extended Euclid's Algorithm
10 min

📖 **Reading:** Extended Euclid's Algorithm: Code
10 min

📖 **Quiz:** Tile a Rectangle with Squares
1 question

📖 **Reading:** Slides
1 min

Applications

Greatest Common Divisor: Code

Naive algorithm:

```
1 def gcd(a, b):
2     assert a >= 0 and b >= 0 and a + b > 0
3
4     if a == 0 or b == 0:
5         return max(a, b)
6
7     for d in range(min(a, b), 0, -1):
8         if a % d == 0 and b % d == 0:
9             return d
10
11     return 1
12
13 print(gcd(0, 1))
14 print(gcd(24, 16))
15 # The following call would take too long
16 #print(gcd(790933790547, 1849639579327))
```

Run

Reset

1
8

Euclid's algorithm, slow implementation:

```
1 def gcd(a, b):
2     assert a >= 0 and b >= 0 and a + b > 0
3
4     while a > 0 and b > 0:
5         if a >= b:
6             a = a - b
7         else:
8             b = b - a
9
10    return max(a, b)
11
12
13 print(gcd(24, 16))
14 print(gcd(790933790547, 1849639579327))
15 # The following call would take too long
16 #print(gcd(790933790548, 2))
17
18
```

Run

Reset

8
3416723

Euclid's algorithm, efficient implementation:

```
1 def gcd(a, b):
2     assert a >= 0 and b >= 0 and a + b > 0
3
4     while a > 0 and b > 0:
5         if a >= b:
6             a = a % b
7         else:
8             b = b % a
9
10    return max(a, b)
11
12
13 print(gcd(24, 16))
14 print(gcd(790933790547, 1849639579327))
15 print(gcd(790933790548, 2))
```

Run

Reset

8
3416723
2

✔ Completed Go to next item