```python
def euclidean_algorithm(a, b):
    while b != 0:
        a, b = b, a % b
    return a

# Пример использования
a = 12345
b = 54321
print(f"НОД({a}, {b}) = {euclidean_algorithm(a, b)}")
```

→ НОД(12345, 54321) = 3

```python
def binary_euclidean_algorithm(a, b):
    g = 1
    while a % 2 == 0 and b % 2 == 0:
        a = a // 2
        b = b // 2
        g = g * 2
    while a != 0:
        while a % 2 == 0:
            a = a // 2
        while b % 2 == 0:
            b = b // 2
        if a >= b:
            a = a - b
        else:
            b = b - a
    return g * b

# Пример использования
a = 12345
b = 54321
print(f"НОД({a}, {b}) = {binary_euclidean_algorithm(a, b)}")
```

→ НОД(12345, 54321) = 3

```python
def extended_euclidean_algorithm(a, b):
    r0, r1 = a, b
    x0, x1 = 1, 0
    y0, y1 = 0, 1
    while r1 != 0:
        q = r0 // r1
        r0, r1 = r1, r0 - q * r1
        x0, x1 = x1, x0 - q * x1
        y0, y1 = y1, y0 - q * y1
    return r0, x0, y0

# Пример использования
a = 12345
b = 54321
d, x, y = extended_euclidean_algorithm(a, b)
print(f"НОД({a}, {b}) = {d}, коэффициенты: x = {x}, y = {y}")
```

    НОД(12345, 54321) = 3, коэффициенты: x = 3617, y = -822

```python
def extended_binary_euclidean_algorithm(a, b):
    g = 1
    while a % 2 == 0 and b % 2 == 0:
        a = a // 2
        b = b // 2
        g = g * 2
    u, v = a, b
    A, B, C, D = 1, 0, 0, 1
    while u != 0:
        while u % 2 == 0:
            u = u // 2
            if A % 2 == 0 and B % 2 == 0:
                A, B = A // 2, B // 2
            else:
                A, B = (A + b) // 2, (B - a) // 2
        while v % 2 == 0:
            v = v // 2
            if C % 2 == 0 and D % 2 == 0:
                C, D = C // 2, D // 2
            else:
                C, D = (C + b) // 2, (D - a) // 2
        if u >= v:
            u, A, B = u - v, A - C, B - D
        else:
            v, C, D = v - u, C - A, D - B
    d = g * v
    x, y = C, D
    return d, x, y

# Пример использования
a = 12345
b = 54321
d, x, y = extended_binary_euclidean_algorithm(a, b)
print(f"НОД({a}, {b}) = {d}, коэффициенты: x = {x}, y = {y}")
```

    НОД(12345, 54321) = 3, коэффициенты: x = -14490, y = 3293