MDM Lista 4

Weronika Jakimowicz

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
X	X	–	X	–	–	X	X	–	–	–	–	–	–	–

ZAD 1.

Na początku warto zauważyć, że

$$\mathrm{lcm}(n, m) = \frac{nm}{\gcd(n, m)}.$$

Jeśli liczby są wystarczająco duże, może okazać się, że iloczyn nm przekracza górny zakres liczb całkowitych języka z jakiego korzystamy. Żeby temu zapobiedz, możemy podzielić większą z nich przez gcd(n, m) i dopiero wynik pomnożyć przez mniejszą z liczb.
Algorytm napisany w języku Python.

```python
# funkcja obliczajaca gcd na podstawie algorytmu Euklidesa
def gcd(n, m):
    if m == 0: return n
    else: return gcd(m, n % m)


def lcm(n, m):
    div = gcd(n, m)

    # wybranie wiekszej z liczb n,m
    mx = m
    mn = n

    if n > m:
        mn = m
        mx = n

    # dziele wieksza liczbe, zeby na pewno po pomnozeniu nie wyjsc poza zakres
    mx = mx / div

    # tak naprawde zwracam (n*m)/gcd(n,m)
    return mn * mx
```

ZAD 2.

Zauważmy, że

$$\gcd(a, b, c, d) = \gcd(\gcd(a, b), c, d) = \gcd(\gcd(a, b), \gcd(c, d))$$

czyli listę liczb całkowitych m_i możemy za każdym razem dzielić na pół aż dojdziemy do momentu kiedy mamy listy 2 lub 1 elementowe. Zakładamy, że $\gcd(a) = a$.
Poniższy algorytm, napisany w Pythonie, jest analogiczny do merge sort, gdzie dzielimy listę na podlisty o podobnym rozmiarze i wykonujemy na nich operacje, po czym łączymy je z powrotem w całość.

```python
# implementacja algorytmu Euklidesa jak w Zad 1.
def euclid(n, m):
    if m == 0: return n
    else: return euclid(m, n % m)

def gcd(k, M):
```

```python
    if k == 1: return M[0] # gcd(a) = a
    if k == 2: return euclid(M[0], M[1]) # mamy liste dwuelementowa

    else:
        # na poczatku rozbijamy liste na dwie podlisty: L i R
        L = []
        R = []

        i = 0
        while i < k//2:
            L.append(M[i])
            i += 1
        while i < k:
            R.append(M[i])
            i += 1

        # gcd(M) to gcd(L, R) - analogicznie jak w merge sort
        return gcd(2, [gcd(k//2, L), gcd(k-k//2, R)])
```

ZAD 3.

Zacznijmy od $k = 2$. Szukamy x_1, x_2 takich, że

$$x_1 m_1 + x_2 m_2 = \gcd(m_1, m_2).$$

Problem ten rozwiązuje rozszerzony algorytm Euklidesa.

ZAD 4.

Jeśli a, b są parzyste, to obie są podzielne przez 2, więc 2 wystąpi co najmniej raz w rozkładzie $\gcd(a, b)$ na czynniki pierwsze. Możemy więc zapisać, że istnieją $a', b' \in \mathbb{N}$ takie, że $a = 2a', b = 2b'$ oraz

$$\gcd(a, b) = 2\gcd(a', b').$$

```python
def binary_gcd(a, b):
    if a == 0:
        return b

    # XOR sprawdza, czy tylko jedna liczba jest parzysta
    bol = (a % 2 == 0) ^ (b % 2 == 0)

    if bol:
        if a % 2 == 1: return binary_gcd(b, a)
        return binary_gcd(a//2, b)

    # jezeli obie sa nieparzyste
    elif (a+b) % 2 == 0:
        if a > b: return binary_gcd(a-b, b)
        return binary_gcd(b-a, a)

    # jezeli obie sa parzyste, to na pewno obie dzieli 2, wiec mozna zwrocic 2 * gcd(
    a/2, b/2)
    else:
        return 2 * binary_gcd(a//2, b//2)
```

Zauważmy, że jeśli chociaż jedna z liczb a, b jest parzysta, to zmniejszamy je 2 razy. Jeżeli obie są nieparzyste, to $a - b$ jest parzyste i wtedy co drugi krok zmniejszamy liczby o połowę. Czyli co najwyżej co dwa obroty zmniejszamy liczby dwukrotnie, więc wykona się $2\log_2 \max(a, b)$ operacji, co daje nam złożoność $O(\log_2 \max(a, b))$.

ZAD 5.

ZAD 6.

Niech $q_1, \ldots, q_k$ będą kolejnymi liczbami pierwszymi takimi, że $q_k = p$. Wtedy

$$m = (m_1, m_2, \ldots, m_k)_p = \sum_{i=1}^{k} m_i q_i$$

ZAD 7.

a) $xz \equiv yz \mod mz \iff x \equiv y \mod m$ dla $z \neq 0$

$\Longleftarrow$

Skoro $x \equiv y \mod m$, to znaczy, że istnieje $k \in \mathbb{Z}$ takie, że

$$x = km + y.$$

Jeżeli teraz pomnożymy obie strony przez z, to dostaniemy

$$xz = kmz + yz.$$

To oznacza, że xz jest podzielne przez mz reszta yz, czyli

$$xz \equiv yz \mod mz.$$

$\Longrightarrow$

Jeżeli $xz \equiv yz \mod mz$, to dla pewnego $k \in \mathbb{Z}$ zachodzi

$$xz = kmz + yz$$

skoro $z \neq 0$, to możemy obustronnie podzielić przez z

$$x = km + y.$$

Z tego z kolei wynika, że

$$x \equiv y \mod m$$

b) $xz \equiv yz \mod m \iff x \equiv y \mod \frac{m}{\gcd(z,m)}$ dla $x, y, z, m \in \mathbb{Z}$

$\Longleftarrow$

Z założenia, że $x \equiv y \mod \frac{m}{\gcd(z,m)}$, to istnieje $k \in \mathbb{Z}$ takie, że

$$x = k\frac{m}{\gcd(z, m)} + y.$$

Jeżeli pomnożymy obie strony przez z, to dostaniemy

$$xz = km\frac{z}{\gcd(z, m)} + yz$$

I zauważmy, że $\frac{z}{\gcd(z,m)} \in \mathbb{Z}$, czyli istnieje $p \in \mathbb{Z}$ takie, że

$$xz = pm + yz$$

a więc

$$xz \equiv yz \mod m$$

$\Longrightarrow$

Zakładamy, że $xz \equiv yz \mod m$, czyli istnieje $k \in \mathbb{Z}$ takie, że

$$xz = km + yz$$

Podzielmy teraz obustronnie przez $\gcd(z, m)$, dostajemy:

$$x\frac{z}{\gcd(z, m)} = \frac{k}{\gcd(z, m)}m + y\frac{z}{\gcd(z, m)}.$$

Zauważmy, że istnieje pewno $p \in \mathbb{Z}$ takie, że

$$z = p \cdot \gcd(z, m)$$

oraz $\gcd(p, m) = 1$. Dalej mamy

$$xp = k\frac{m}{\gcd(z, m)} + yp \quad (☕)$$
$$p(x - y) = k\frac{m}{\gcd(z, m)}$$

Ponieważ prawa strona równania jest liczbą całkowitą podzielną przez p, to również lewa strona musi być podzielna przez p. Zauważmy, że $p \nmid m$, a więc również $p \nmid \frac{m}{\gcd(z,m)}$. W takim razie $p \mid k$ i wtedy $\frac{k}{p} \in \mathbb{Z}$. Czyli jeśli podzielimy obie strony równania (☕) przez p, dostajemy

$$x = \frac{k}{p}\frac{m}{\gcd(z, m)} + y$$

i z tego wynika, że

$$x \equiv y \mod \frac{m}{\gcd(z, m)}$$

c) $x \equiv y \mod mz \implies x \equiv y \mod m$

Ponieważ zwykle operacja modulo n jest zdefiniowana dla n całkowitych, założę, że jednocześnie mz jak i m są liczbami całkowitymi. Dodatkowo, m jest liczbą pierwszą, bo w przeciwnym przypadku mamy kontrprzykład dla $m = 18$, $z = \frac{1}{3}$ i $x = 10$:

$$10 \equiv 4 \mod 6$$

$$10 \equiv 10 \mod 18$$

Jeżeli $z \in \mathbb{Z}$, to istnieje $k \in \mathbb{Z}$ takie, że $\gcd(a, b) = 1$ oraz

$$x = kmz + y$$

i wtedy również $kz \in \mathbb{Z}$, czyli $x \equiv y \mod m$.
W przeciwnym wypadku $z \in \mathbb{Q}$, a więc istnieje $a \in \mathbb{Z}$ oraz $b \in \mathbb{N}$ takie, że

$$z = \frac{a}{b}.$$

I ponieważ m jest liczbą pierwszą, to $\gcd(m, b) = m$. Istnieje $p \in \mathbb{Z}$ takie, że $pm = b$ i jeżeli $p \neq 1$, to mamy

$$mz = m\frac{a}{b} = m\frac{a}{pm} = \frac{a}{p} \in \mathbb{Z}$$

co daje sprzeczność z $\gcd(a, b) = 1$, więc $p = 1$ i $b = m$.

$$x = kmz + y$$
$$x = km\frac{a}{m} + y$$
$$xm = kam + ym$$

więc $xm \equiv ym \mod m$. Z poprzedniego podpunktu mamy

$$x \equiv y \mod \frac{m}{\gcd(m, m)} = x \equiv y \mod 1$$

czyli $y = 0$, więc $mz|x$ i $m|x$ i mamy $x \equiv 0 \mod m$.

ZAD 8.

a) $2^n - 1$ – liczba pierwsza $\implies$ n jest liczbą pierwszą

Załóżmy nie wprost, że istnieje n takie, że n nie jest liczbą pierwszą, ale $2^n - 1$ jest liczbą pierwszą. Ponieważ n nie jest pierwsze, to istnieją $k, m \in \mathbb{Z}$ większe od 1 takie, że $n = km$. Wtedy

$$2^n - 1 = 2^km - 1 = (2^k)^m - 1 = (2^k - 1)(2^{k(m-1)} + 2^{k(m-2)} + ... + 2^1 + 2^0)$$

Ponieważ $2^k > 2^1 = 2$, to mnożymy liczbę całkowitą $(2^{k(m-1)}+2^{k(m-2)}+...+2^1+2^0)$ przez liczbę całkowitą różną od 1 (2^k-1). W takim razie 2^n-1 nie jest liczbą pierwszą i mamy sprzeczność.

b) $a^n - 1$ jest liczbą pierwszą, to $a = 2$

Wiemy, że $a^n - 1$ jest liczbą pierwszą, czyli nie ma dzielników całkowitych. Podobnie jak w poprzednim podpunkcie, rozpiszmy wyrażenie jako iloczyn sum:

$$a^n - 1 = (a-1)(a^{n-1} + a^{n-2} + ... + a^1 + a^0).$$

W takim razie $a - 1$ musi być równe 1, wtedy $a = 2$, lub

$$a^{n-1} + a^{n-2} + ... + a + 1 = 1,$$

a więc

$$a^{n-1} + a^{n-2} + ... + a = 0$$

co dawałoby $a = 0$, co nie jest poprawne, bo $0^n - 1 = -1$ nie jest liczbą pierwszą.

c) $2^n + 1$ – liczba pierwsza $\implies (\exists k)\, 2^k = n$

Załóżmy nie wprost, że n nie jest potęgą liczby 2. W takim razie $n = km$ gdzie $k, m \in \mathbb{N}$ i m jest liczbą nieparzystą.

$$2^n + 1 = 2^{km} + 1 = (2^k)^m + 1$$

niech $a = 2^k$, wtedy

$$a^x + 1 = a^x - (-1)^x = (a+1)(a^{x-1} + a^{x-2} + ... + a + 1)$$

Czyli 2^n+1 jest iloczynem $(a+1) = 2^y+1$ gdzie $y > 1$, więc 2^n+1 jest iloczynem dwóch liczb całkowitych i nie może być liczbą pierwszą.