# ① ZBIORY, ODWZOROWANIA I RELACJE

<u>Aksjomaty</u>: 1) Jeśli zbiory $A$ i $B$ mają te same elementy to są identyczne.

2) $\forall\, A$ i $B$ $\exists\, A \cup B$

3) $\exists$ conajmniej jeden zbiór

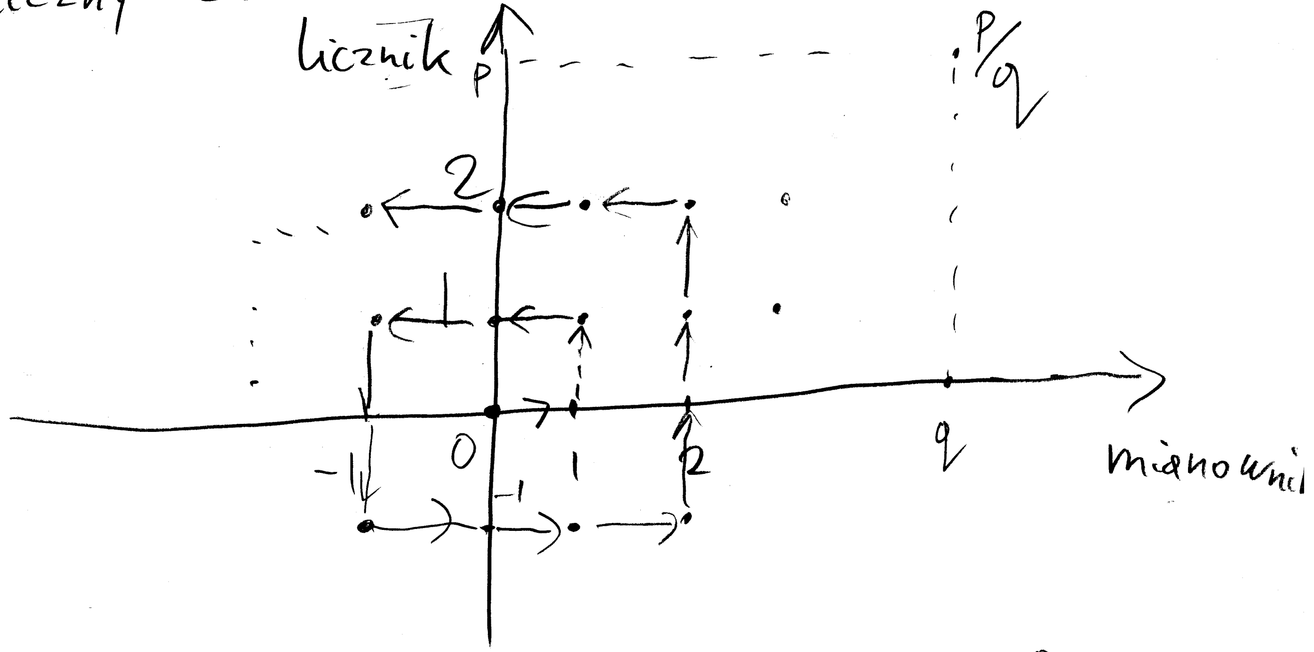
4) $\forall\, A$ i $\varphi$ $\exists\, \{x \in A \mid \varphi(x)$ jest prawdziwe$\}$

5) $\forall\, A\ \exists\, 2^A$ 6) Aksjomat wyboru.

<u>Definicja</u>: Zbiory $X$ i $Y$ są równoliczne jeśli $\exists\, X \xrightarrow{f} Y$ : ① $\forall y \in Y\ \exists x \in X: f(x) = y$

② $f(x) = f(x') \Longrightarrow x = x'$.

<u>Przykład</u>: Zbiór liczb naturalnych $\mathbb{N}$ jest równoliczny ze zbiorem liczb wymiernych $\mathbb{Q}$:




Wyrzucając $\frac{p}{0}$ dla $p \neq 0$ oraz $\frac{p'}{q'}$ jeśli $\frac{p'}{q'} = \frac{p}{q}$ i $\frac{p}{q}$ już mamy, otrzymujemy bijekcję $\mathbb{N} \ni n \mapsto$ n-ty elt $\in \mathbb{Q}$ na oczyszczonej spirali ∎

<u>Przykład</u>: $X = \{1, \dots, n\}$ i $2^X$ = zbiór wszystkich podzbiorów $X$ <u>nie</u> są równoliczne bo $|X| = n$ a $|2^X| = 2^n$.
Istotnie, weźmy $x \in X$, $A \subseteq X$. Zdefiniujmy

$$2^X \xrightarrow{f_x} \{0, 1\}$$

$$f_x(A) := \begin{cases} 0 & \text{jeśli } x \notin A \\ 1 & \text{jeśli } x \in A \end{cases}$$

Znajomość $A$ to znajomość wszystkich wartości wszystkich funkcji $f_x$:
$(f_1(A), f_2(A), \dots, f_n(A))$. Tyle jest $A \in 2^X$ ile jest wszystkich $n$-ciągów o wyrazach z $\{0, 1\}$, czyli $2^n$.

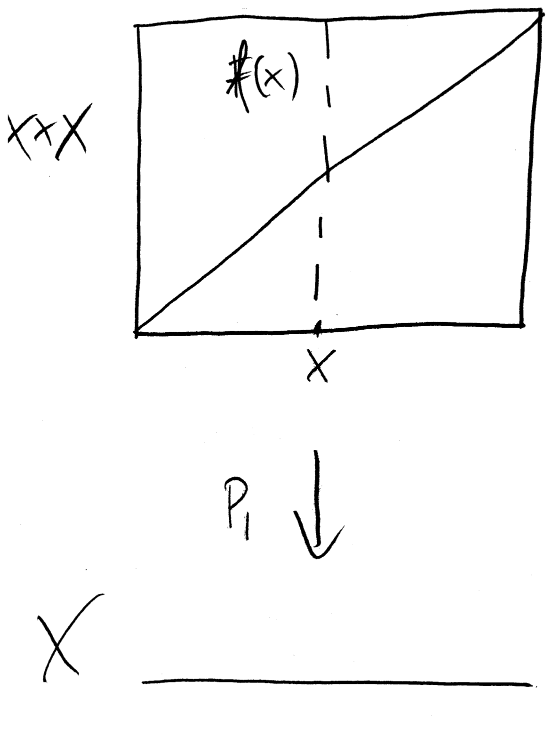
<u>Twierdzenie Cantora</u>: **Dla dowolnego zbioru $X$ <u>nie</u> istnieje jakakolwiek surjekcja z $X$ na zbiór wszystkich podzbiorów $2^X$.**

<u>Dowód</u>: Niech $X \xrightarrow{f} 2^X$ będzie dowolnym odwzorowaniem i $Y := \{x \in X \mid x \notin f(x)\}$. Wtedy
$\forall x \in X: Y \neq f(x)$ bo inaczej $x_0 \in Y = f(x_0) \Leftrightarrow x_0 \notin f(x_0)$
Stąd $\nexists\, X \twoheadrightarrow 2^X$. $\square$

**Wniosek**: Nie istnieje zbiór wszystkich zbiorów.

**Dowód**: Przypuśćmy że $X$ jest zbiorem wszystkich zbiorów. Wtedy $2^X \subseteq X$.
Stąd $\exists\ X \xrightarrow{f} 2^X : f(x) = \begin{cases} \emptyset & \text{jeśli } x \notin 2^X \\ x & \text{jeśli } x \in 2^X \end{cases}$.
To jest sprzeczne z Twierdzeniem Cantora. $\square$

**Ilustracja Twierdzenia Cantora**:




$$M := \{(x,y) \in X \times X \mid y \in f(x)\}$$

$$D := \{(x,y) \in X \times X \mid y = x\}$$

$$Y = P_1(D \setminus M)$$

**Paradoks Russella**: $R$ zbiór zbiorów takich że $X \notin X$. Wtedy $R \in R \Rightarrow R \notin R$ i $R \notin R \Rightarrow R \in R$.

**Twierdzenie Cantora-Bernsteina**: Jeśli $X \subseteq Y \subseteq Z$ i $|X| = |Z|$, to $|Y| = |X|$.

Relacja równoważności w zbiorze $X$ to podzbiór $R \subseteq X \times X$ spełniający następujące warunki:

① $\forall x \in X: \quad (x,x) \in R$ (zwrotność)

② $(x,y) \in R \iff (y,x) \in R$ (symetryczność)

③ $(x,y),(y,z) \in R \Rightarrow (x,z) \in R$ (przechodniość)

Przykład: $R := \{(x,y) \in X \times X \mid y = x\}$

Przykład: Niech $\{U_i\}_{i \in I}$ będzie rodziną podzbiorów $X$ taką że $\bigcup_{i \in I} U_i = X$. Taką rodzinę nazywamy pokryciem. Zdefiniujmy

$$R = \{(x,y) \in X \times X \mid x \in U_i \iff y \in U_i, \forall i \in I\}$$

Klasy równoważności: $[x]_R := \{y \in X \mid (x,y) \in R\}$

$$[x] \cap [y] \neq \phi \Rightarrow [x] = [y]$$

Istotnie, $\exists z_0 \in [x] \cap [y]$. Stąd $z \in [x] \Rightarrow$

$$(x,z) \in R \xRightarrow{(z_0,x) \in R} (z_0,z) \in R \xRightarrow{(y,z_0) \in R} (y,z) \in R.$$

$\Rightarrow z \in [y]$. Podobnie $z \in [y] \Rightarrow z \in [x]$.

Wniosek: Zbiór $X$ daje się rozłożyć na sumę rozłącznych podzbiorów będących klasami równoważności. Zbiór wszystkich klas równoważności oznaczamy $X/R$ i nazywamy przestrzenią ilorazową.

Przykład: Dla $p \in \mathbb{N}$, niech $R_p = \{(m,n) \in \mathbb{Z} \times \mathbb{Z} \mid m-n = pk, k \in \mathbb{Z}\}$

Dla $p=0$ $\mathbb{Z}/R_p = \mathbb{Z}$, dla $p>0$ $\mathbb{Z}/R_p = \mathbb{Z}/p\mathbb{Z} = \mathbb{Z}/p = \mathbb{Z}_p$

$= \{0, 1, \dots, p-1\}$ liczby modulo $p$.

Zastosowanie: Algorytm kodowania RSA.

$$\boxed{\text{Jeśli } p \text{ i } q \text{ to duże różne liczby pierwsze oraz } de = 1 \bmod (p-1)(q-1), \text{ to } m = m^{de} \bmod pq.}$$

$m$ - wielkość kodowana, $(n,e)$ - klucz publiczny

$(n = pq$, $e$ względnie pierwsze z $(p-1)(q-1))$;

$((p-1)(q-1), d = e^{-1} \bmod (p-1)(q-1))$ - klucz prywatny

Kodujemy $m$ do $c := m^e \bmod n$. Odkodowujemy $m = c^d \bmod n$.

Bezpieczeństwo kodowania RSA opiera się na braku szybkich algorytmów faktoryzujących $n$ na $pq$. Znając $p, q$ i $e$, można wyliczyć $d = e^{-1} \bmod (p-1)(q-1)$.

Takie odwracalne $e$ zawsze istnieje dzięki twierdzeniu że $\forall k > 1$ $\exists$ pierwsza $e$: $k \leq e < 2k$. Istotnie, $(p-1)(q-1) = 2k$, więc $e$ nie dzieli $(p-1)(q-1)$ i jest pierwsza, a zatem odwracalna mod $(p-1)(q-1)$.

<u>Zadanie 1</u>: Przeliczyć przykład kodowania RSA dla $p = 3$, $q = 5$.

<u>Zadanie 2</u>: Udowodnić że suma odwrotności wszystkich liczb pierwszych jest nieskończona.

$p = 61$ and $q = 53$

2. Compute $n = pq$

$$n = 61 \cdot 53 = 3233$$

3. Compute the product of totients. For primes the totient is maximal and equals $x - 1$. Therefore $\varphi(pq) = (p-1)(q-1)$

$$\varphi(61 \cdot 53) = (61-1) \cdot (53-1) = 3120$$

4. Choose any number $e > 1$ that is coprime to 3120. Choosing a prime number for $e$ leaves you with a single check: that $e$ is not a divisor of 3120.

$$e = 17$$

5. Compute $d$ such that $de \equiv 1 \pmod{\varphi(pq)}$ e.g., by computing the modular multiplicative inverse of $e$ modulo $\varphi(pq)$:

$$d = 2753$$

since 17 · 2753 = 46801 and 46801 mod 3120 = 1, this is the correct answer. (iterating finds (15 times 3120)+1 divided by 17 is 2753, an integer, whereas other values in place of 15 do not produce an integer. The extended euclidean algorithm finds the solution to Bézout's identity of 3120x2 + 17x-367=1, and -367 mod 3120 is 2753)

The **public key** is ($n = 3233, e = 17$). For a padded message $m$ the encryption function is $m^{17} \mod 3233$ or abstractly:

$$c = m^e \mod n$$

The **private key** is ($n = 3233, d = 2753$). The decryption function is $c^{2753} \mod 3233$ or in its general form:

$$m = c^d \mod n$$

For instance, in order to encrypt $m = 123$, we calculate

$$c = 855 = 123^{17} \mod 3233$$

To decrypt $c = 855$, we tap

$$m = 123 = 855^{2753} \mod 3233.$$

Both of these calculations can be computed efficiently using the square-and-multiply algorithm for modular exponentiation. In real life situations the primes selected would be much larger, however in our example it would be relatively trivial to factor $n$, 3233, obtained from the freely available public key back to the primes $p$ and $q$. Given $e$, also from the public key, we could then compute $d$ and so acquire the private key.

## Padding schemes