Stefan Sokołowski

# KRYPTOGRAFIA

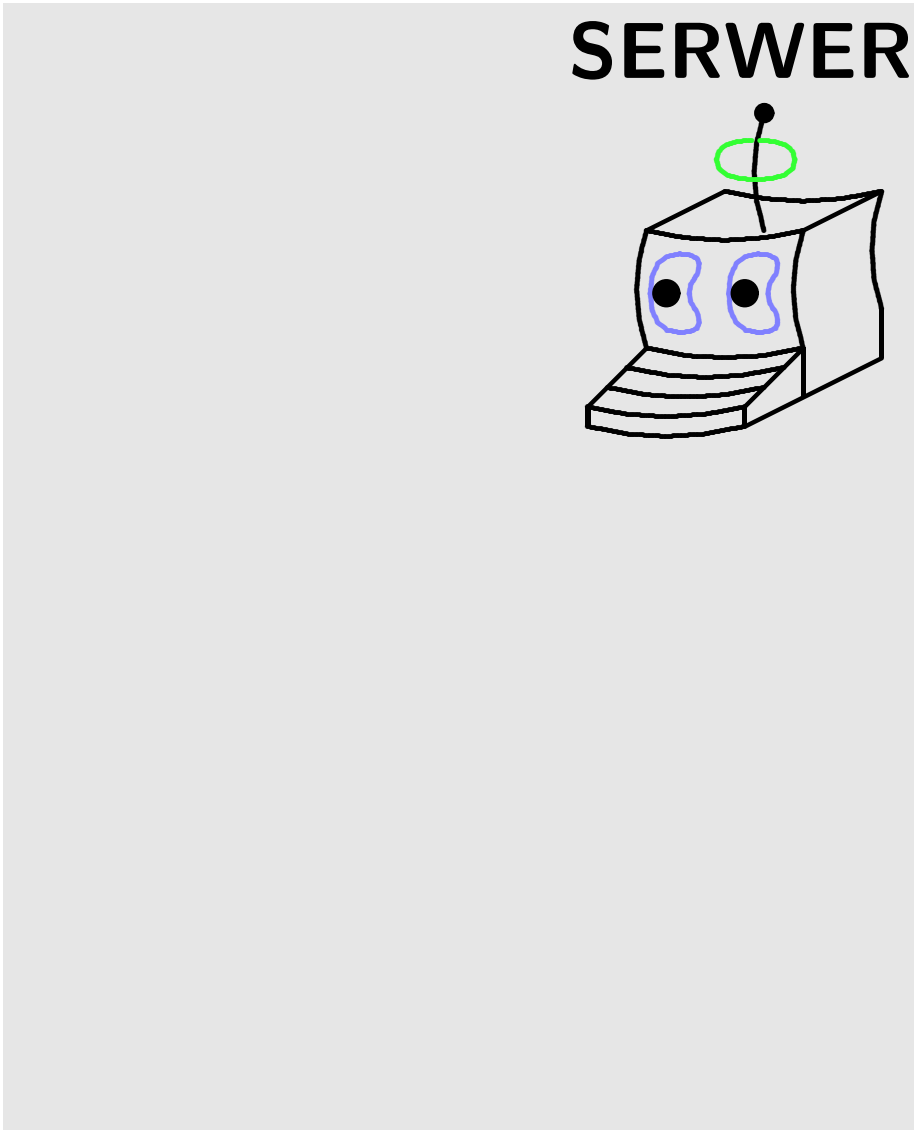
## wykład 5

Inst. Informatyki Stosowanej, ANS
Elbląg, 2026/2027

# Jednokierunkowe funkcje skrótu

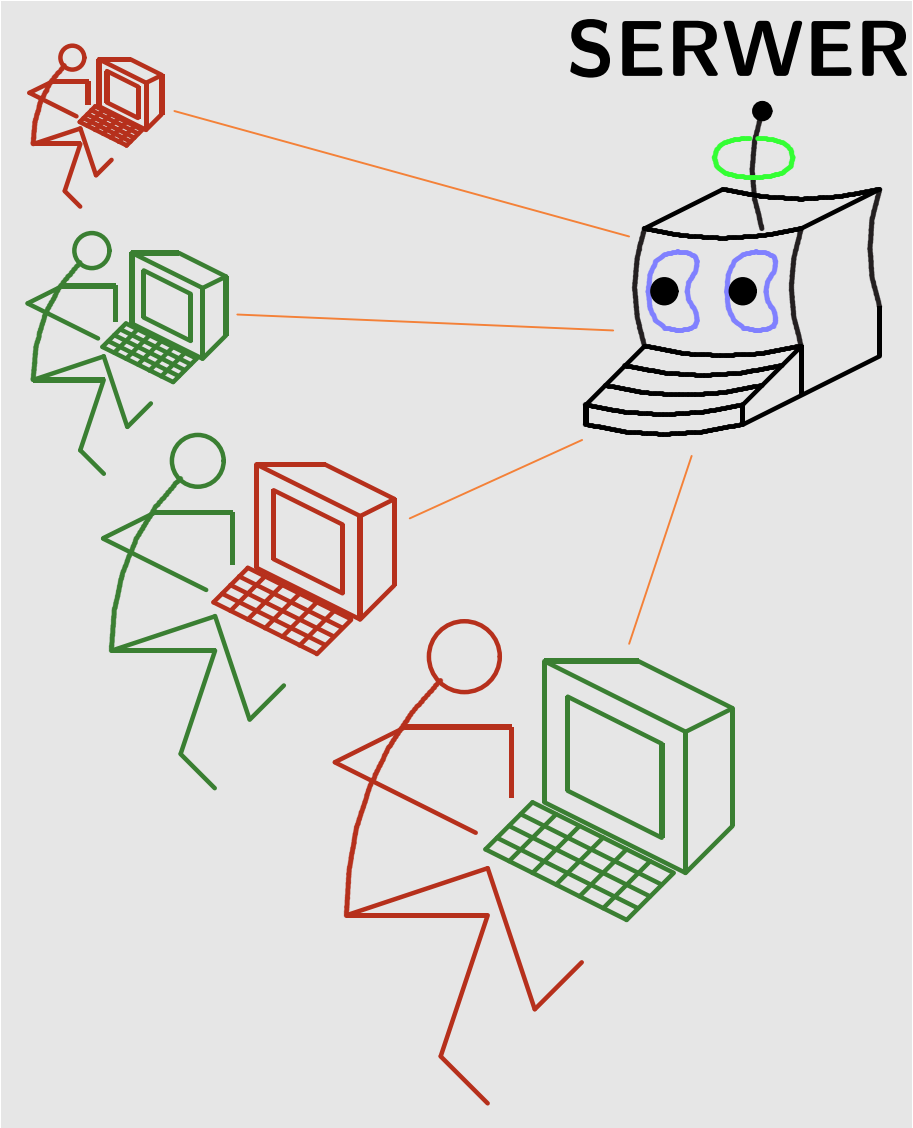
# Jednokierunkowe funkcje skrótu

## Logowanie do serwera:



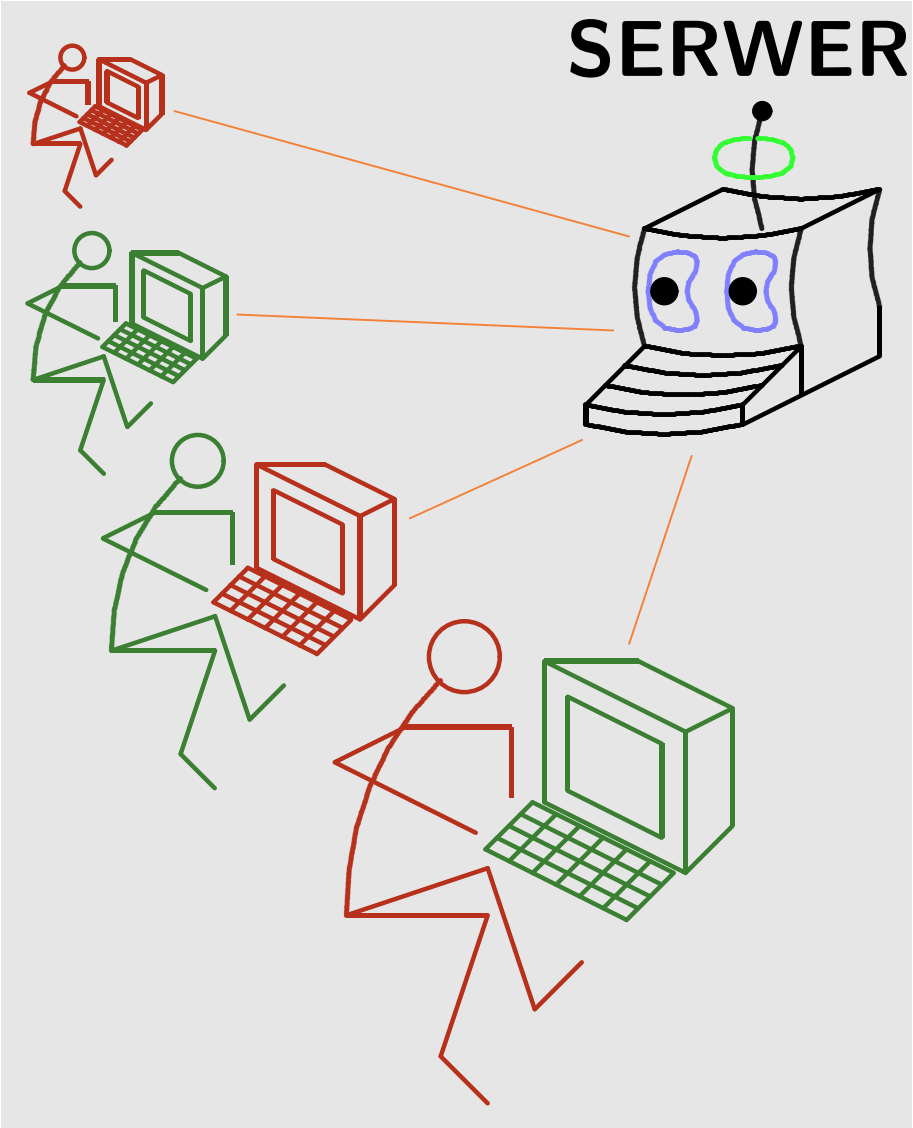
# Jednokierunkowe funkcje skrótu

## Logowanie do serwera:



# Jednokierunkowe funkcje skrótu

## Logowanie do serwera:

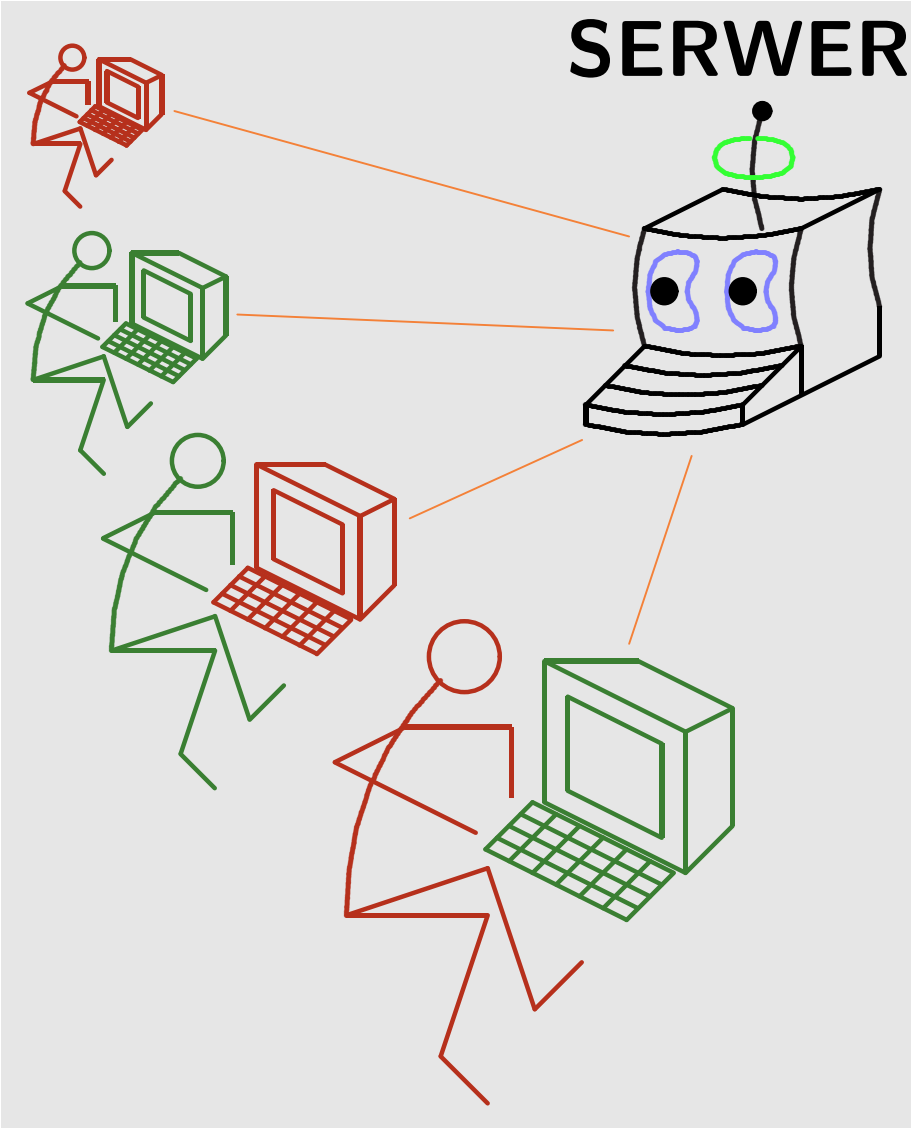



**Na serwerze lista loginów i haseł?**

| LOGIN | PASSWORD |
|---|---|
| Janek | GoOlimpia! |
| Kasia | abd666 |
| Monika | Nobody# |
| Wiesiek | J´aimeKasia |

# Jednokierunkowe funkcje skrótu

## Logowanie do serwera:



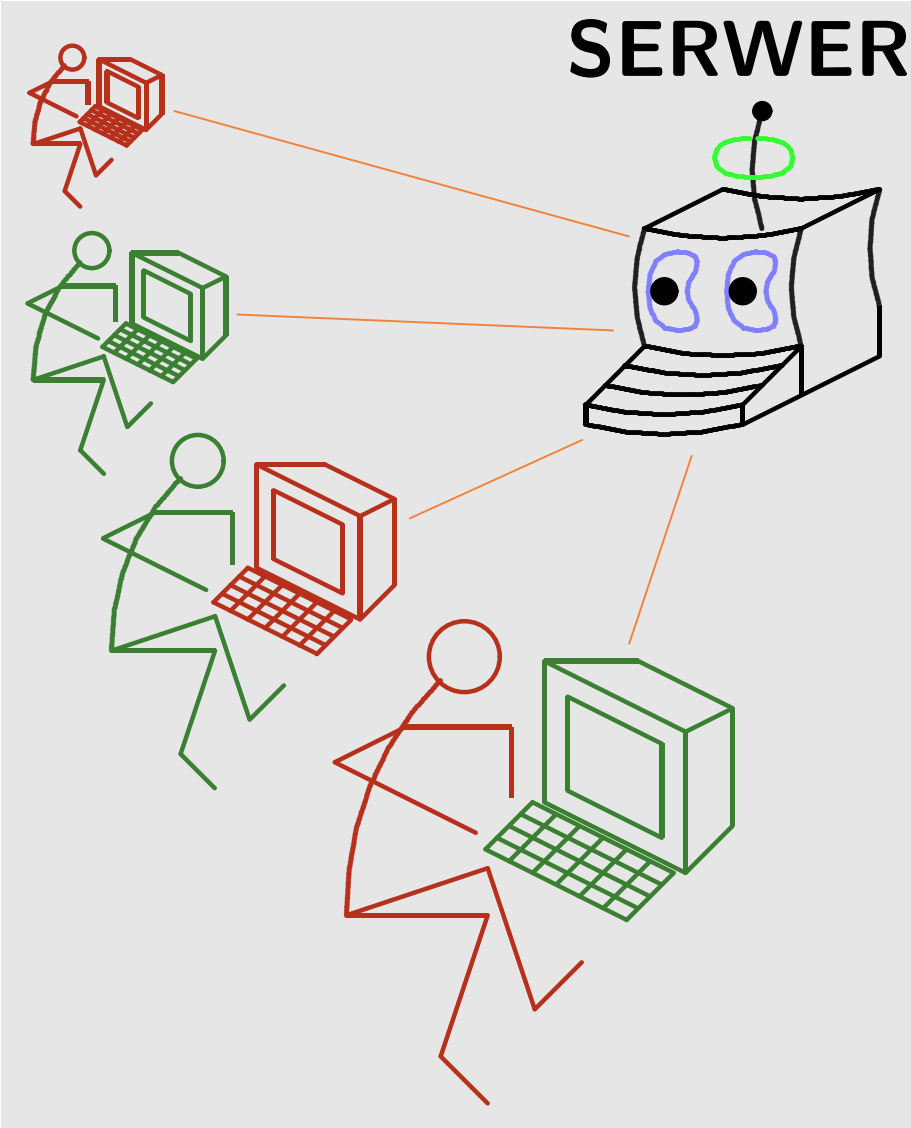

**Na serwerze lista loginów i haseł?**

| LOGIN | PASSWORD |
|---|---|
| Janek | GoOlimpia! |
| Kasia | abd666 |
| Monika | Nobody# |
| Wiesiek | J´aimeKasia |

**To jest niebezpieczne**, bo ktoś może wykraść taką listę i uzyskać dostęp do kont wszystkich użytkowników.

# Jednokierunkowe funkcje skrótu

## Logowanie do serwera:




**Na serwerze lista loginów i haseł?**

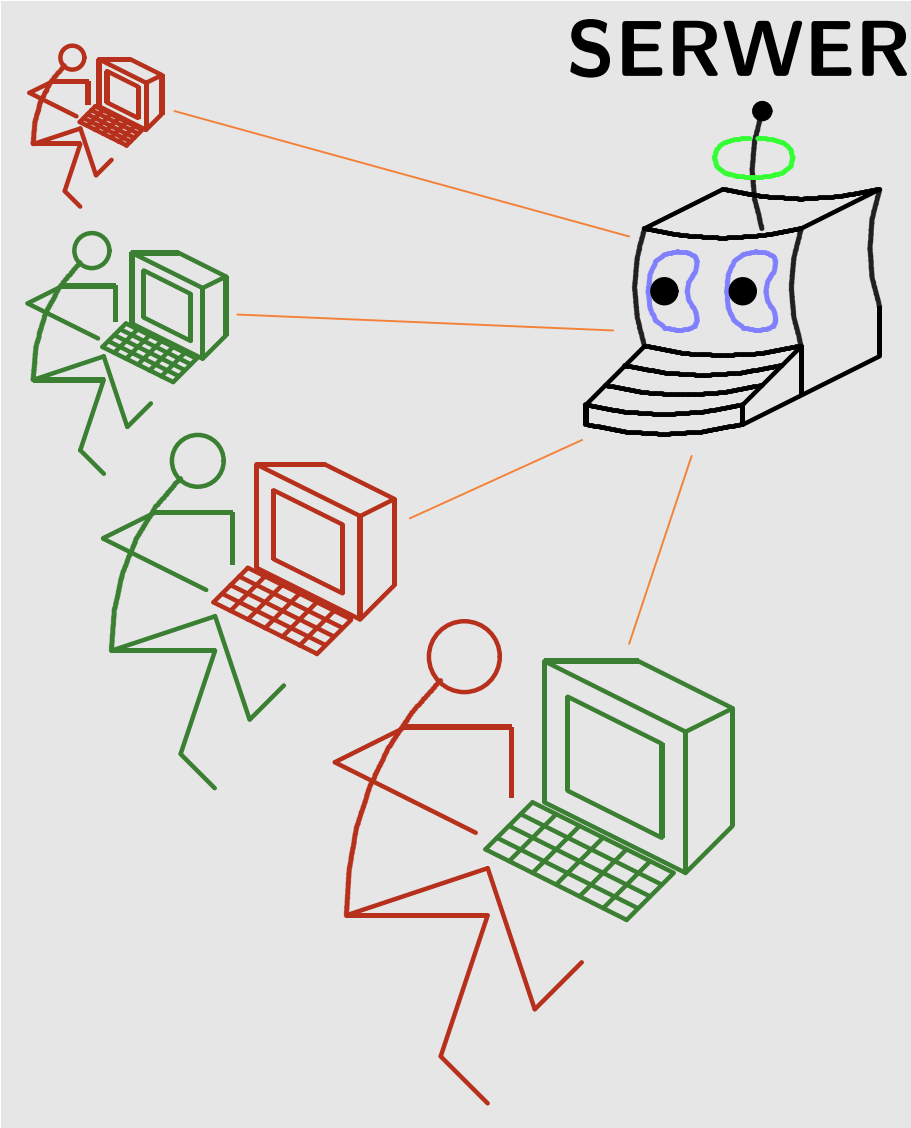
| LOGIN | PASSWORD |
|---|---|
| Janek | GoOlimpia! |
| Kasia | abd666 |
| Monika | Nobody# |
| Wiesiek | J´aimeKasia |

**To jest niebezpieczne**, bo ktoś może wykraść taką listę i uzyskać dostęp do kont wszystkich użytkowników.

Hasła są przekształcane odpowiednią funkcją i zapamiętywane w tej przekształconej formie.

# Jednokierunkowe funkcje skrótu

## Logowanie do serwera:




**Na serwerze lista loginów i haseł?**

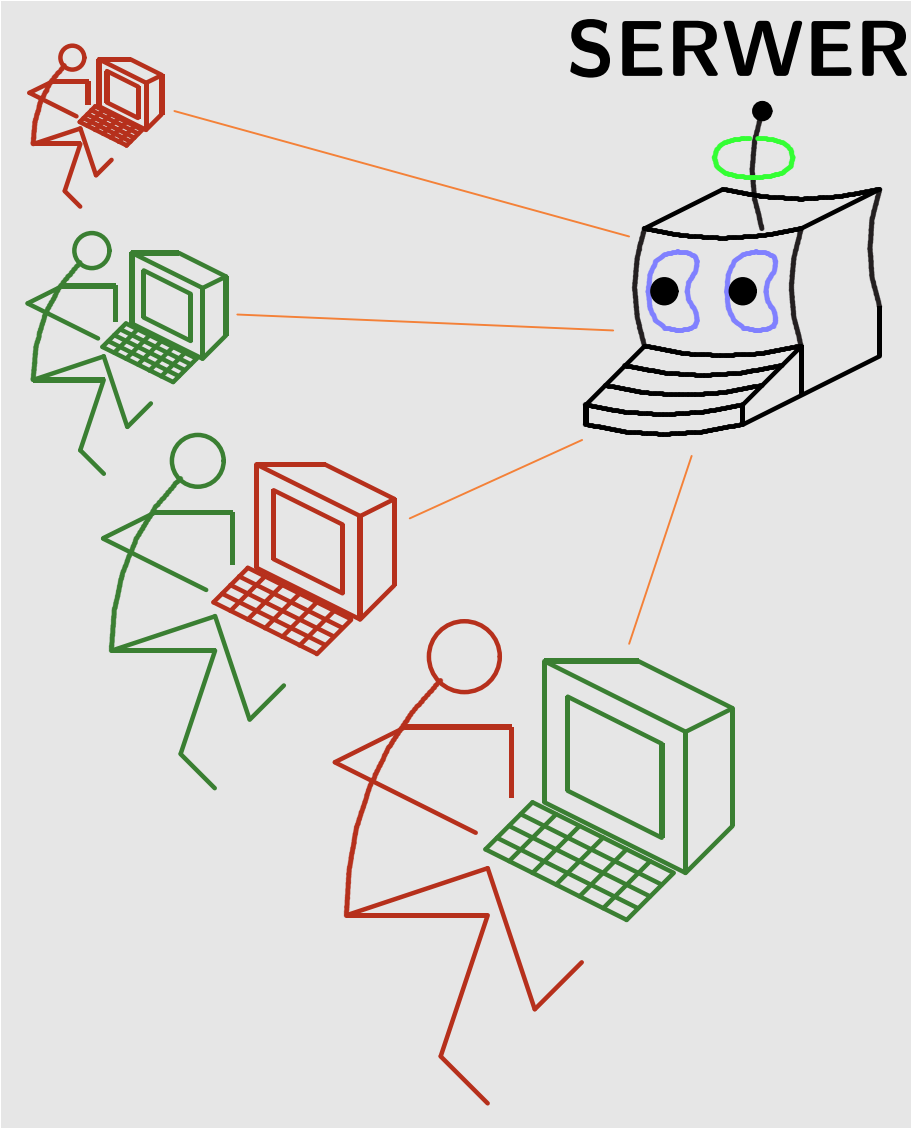
| LOGIN | PASSWORD |
|---|---|
| Janek | GoOlimpia! |
| Kasia | abd666 |
| Monika | Nobody# |
| Wiesiek | J´aimeKasia |

**To jest niebezpieczne**, bo ktoś może wykraść taką listę i uzyskać dostęp do kont wszystkich użytkowników.

Hasła są przekształcane odpowiednią funkcją i zapamiętywane w tej przekształconej formie. Kiedy użytkownik się loguje, podane przez niego hasło jest przekształcane tą samą funkcją

# Jednokierunkowe funkcje skrótu

## Logowanie do serwera:




**Na serwerze lista loginów i haseł?**

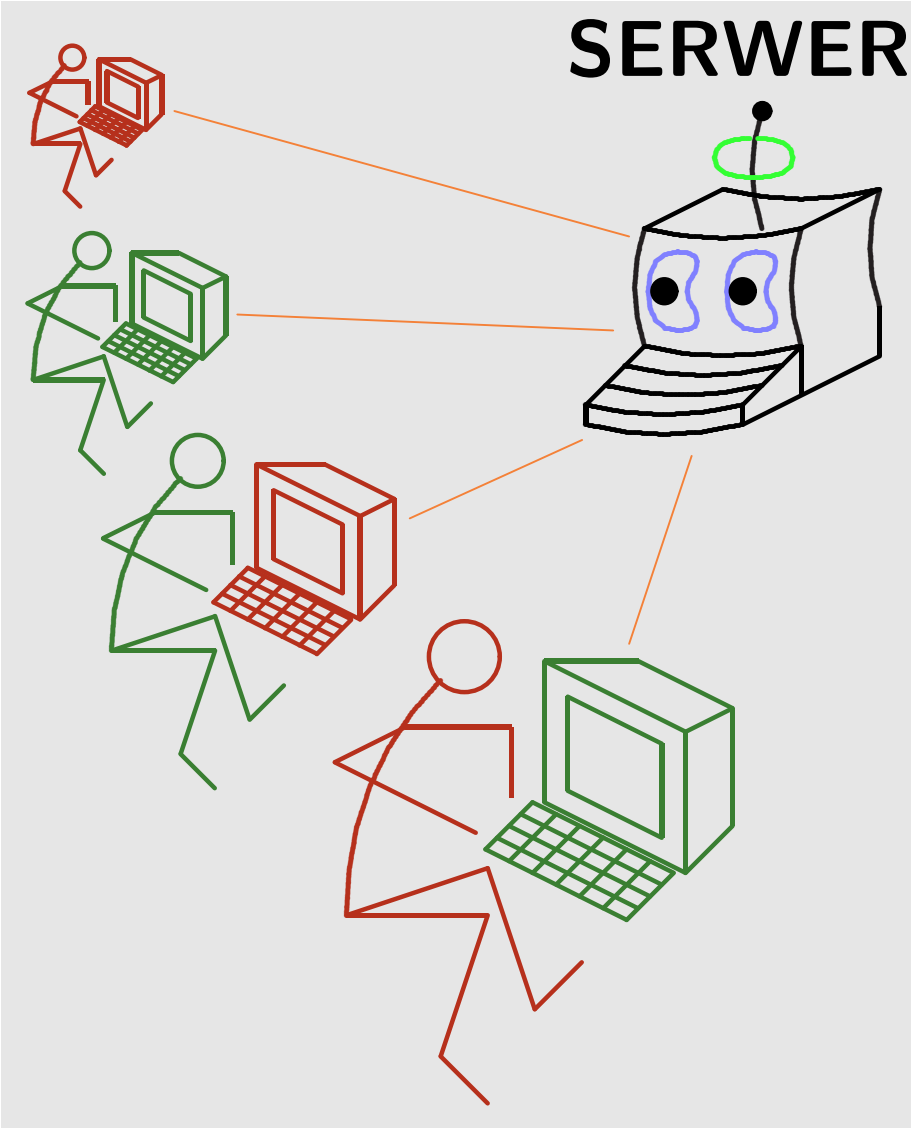
| LOGIN | PASSWORD |
|---|---|
| Janek | GoOlimpia! |
| Kasia | abd666 |
| Monika | Nobody# |
| Wiesiek | J´aimeKasia |

**To jest niebezpieczne**, bo ktoś może wykraść taką listę i uzyskać dostęp do kont wszystkich użytkowników.

Hasła są przekształcane odpowiednią funkcją i zapamiętywane w tej przekształconej formie. Kiedy użytkownik się loguje, podane przez niego hasło jest przekształcane tą samą funkcją i porównywane z zapamiętanym.

# Jednokierunkowe funkcje skrótu

## Logowanie do serwera:




**Na serwerze lista loginów i haseł?**

| LOGIN | PASSWORD |
|---|---|
| Janek | GoOlimpia! |
| Kasia | abd666 |
| Monika | Nobody# |
| Wiesiek | J´aimeKasia |

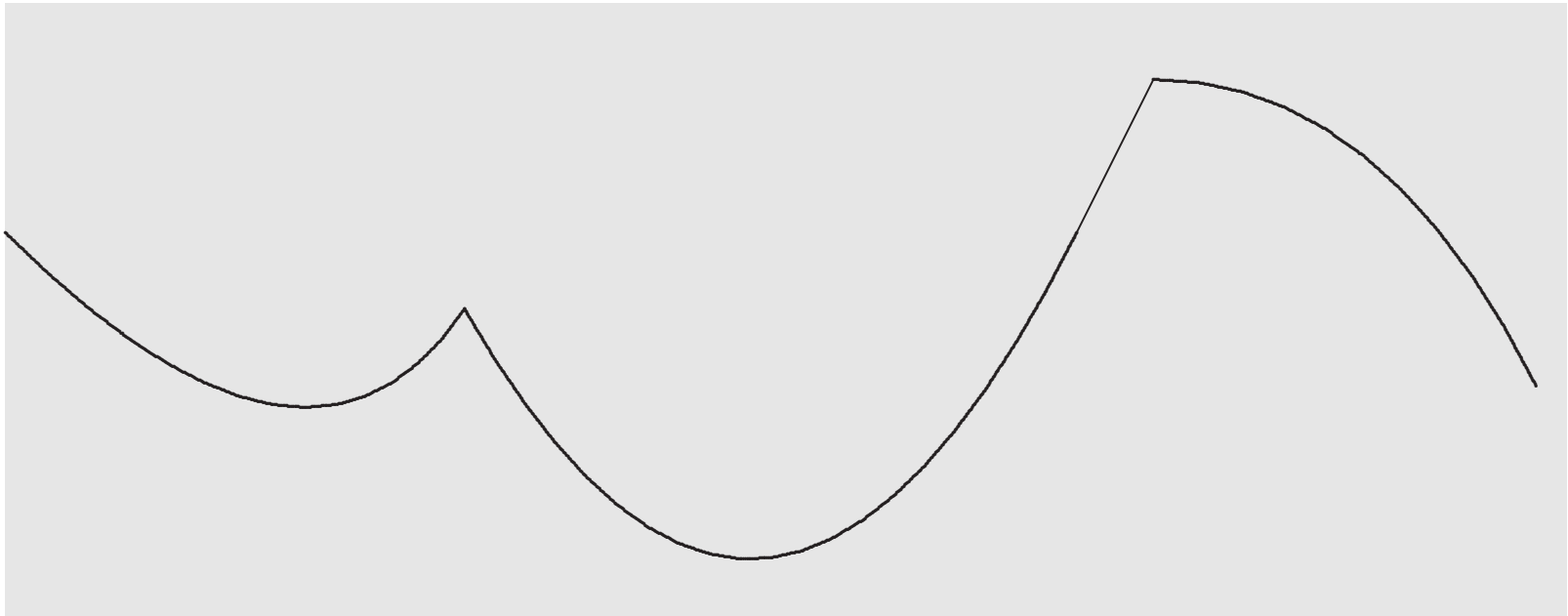
**To jest niebezpieczne**, bo ktoś może wykraść taką listę i uzyskać dostęp do kont wszystkich użytkowników.

Hasła są przekształcane odpowiednią funkcją i zapamiętywane w tej przekształconej formie. Kiedy użytkownik się loguje, podane przez niego hasło jest przekształcane tą samą funkcją i porównywane z zapamiętanym.

**Jaka to może być funkcja?**

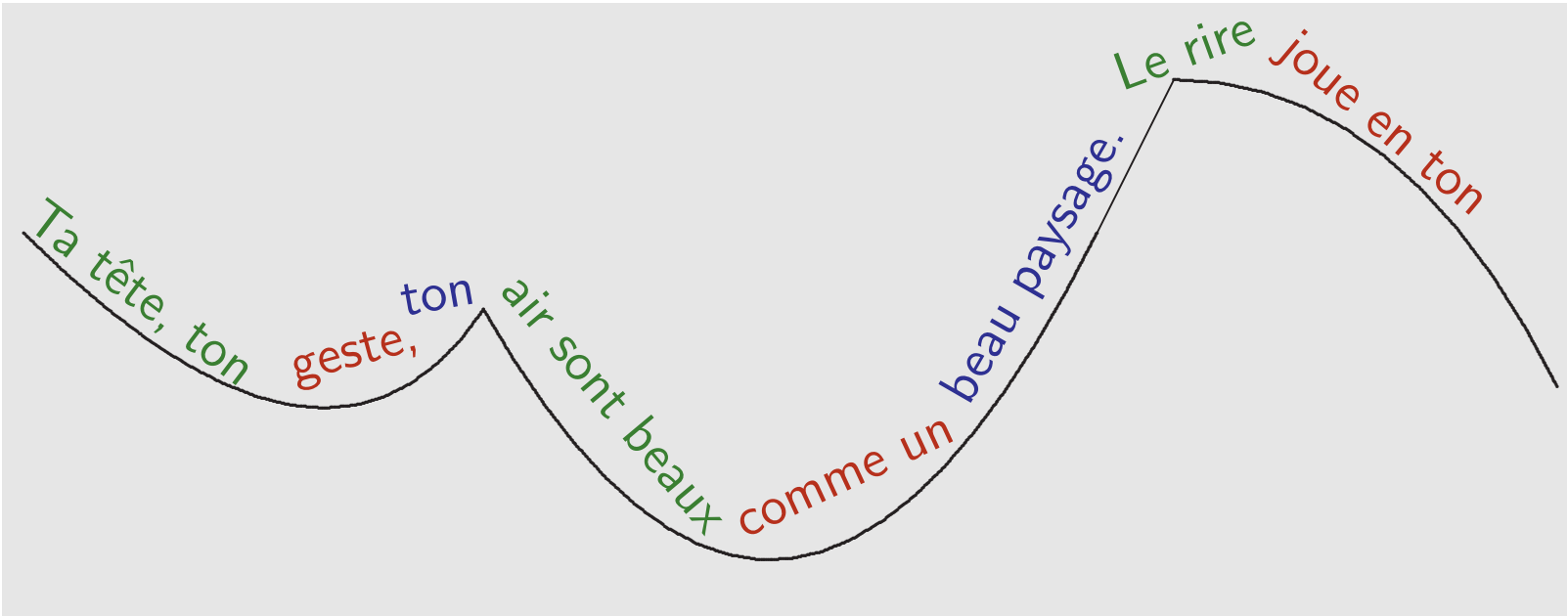
# Jednokierunkowe funkcje skrótu

## Sprawdzanie poprawności transmisji (integralności przekazu):



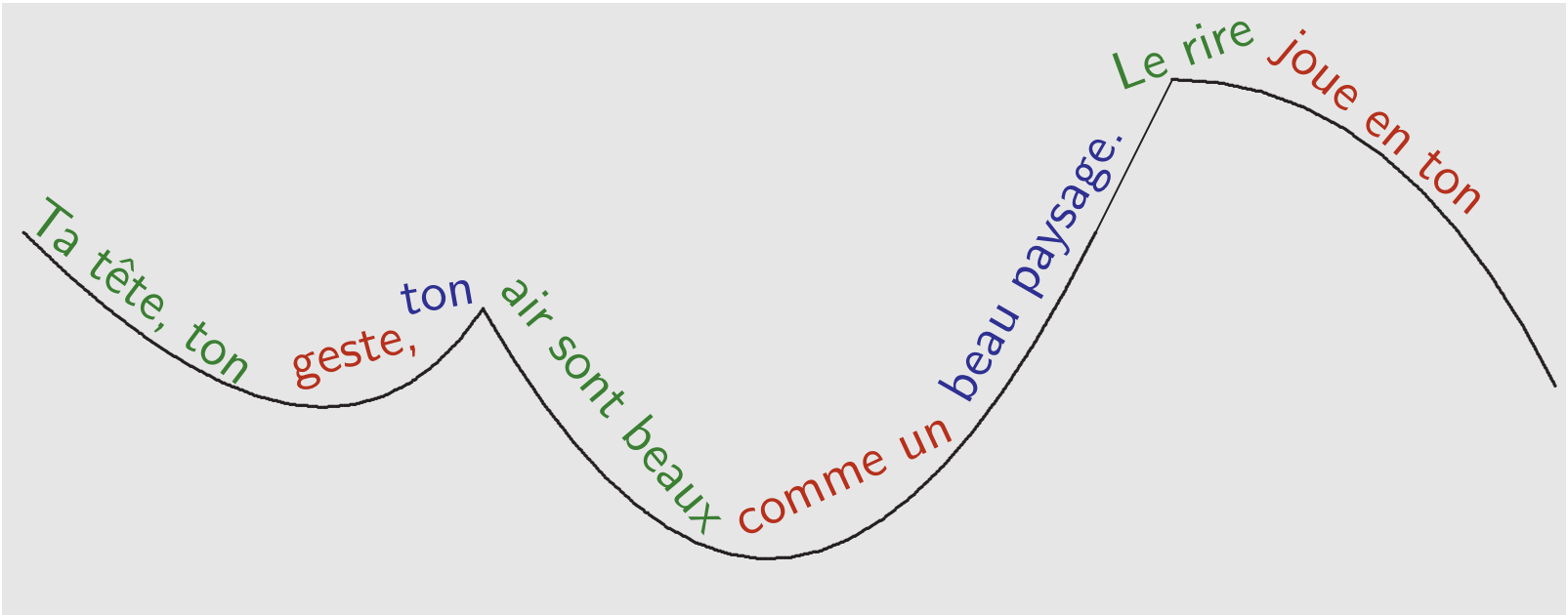
# Jednokierunkowe funkcje skrótu

## Sprawdzanie poprawności transmisji (integralności przekazu):



# Jednokierunkowe funkcje skrótu

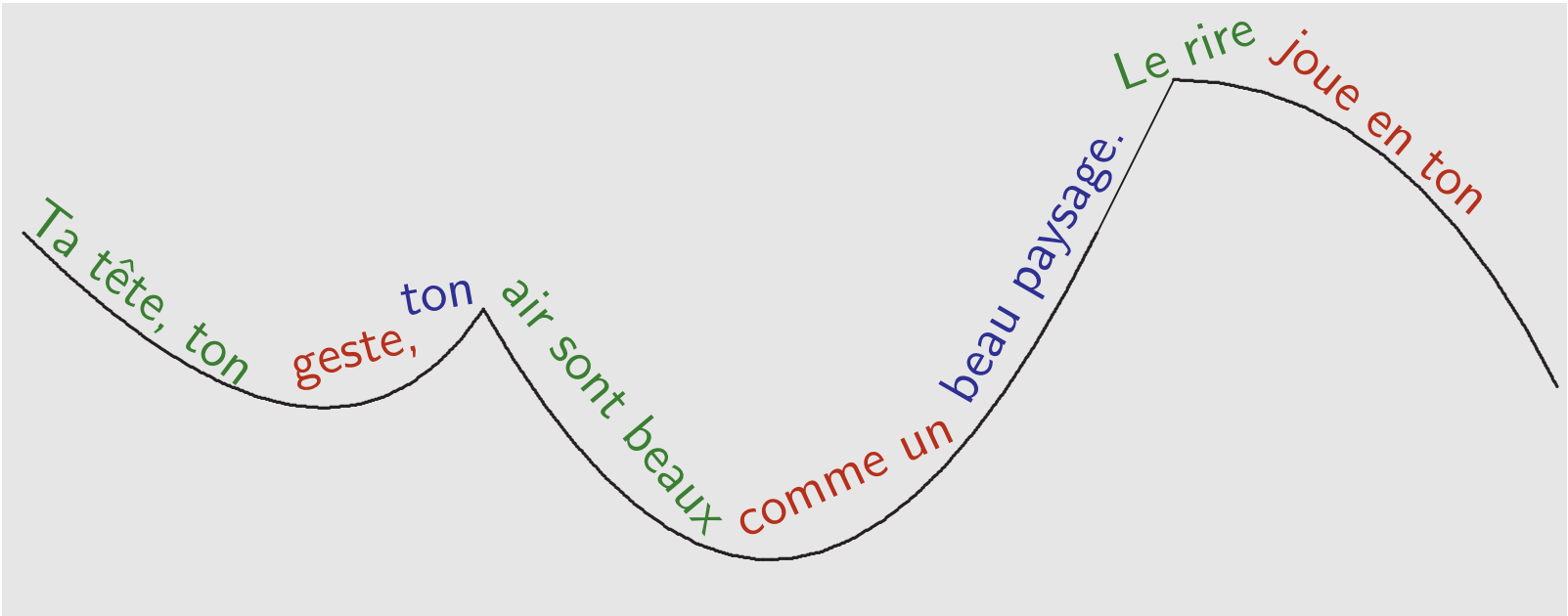
## Sprawdzanie poprawności transmisji (integralności przekazu):




Czy przesyłana wiadomość nie zniekształciła się na wybojach?

# Jednokierunkowe funkcje skrótu

## Sprawdzanie poprawności transmisji (integralności przekazu):

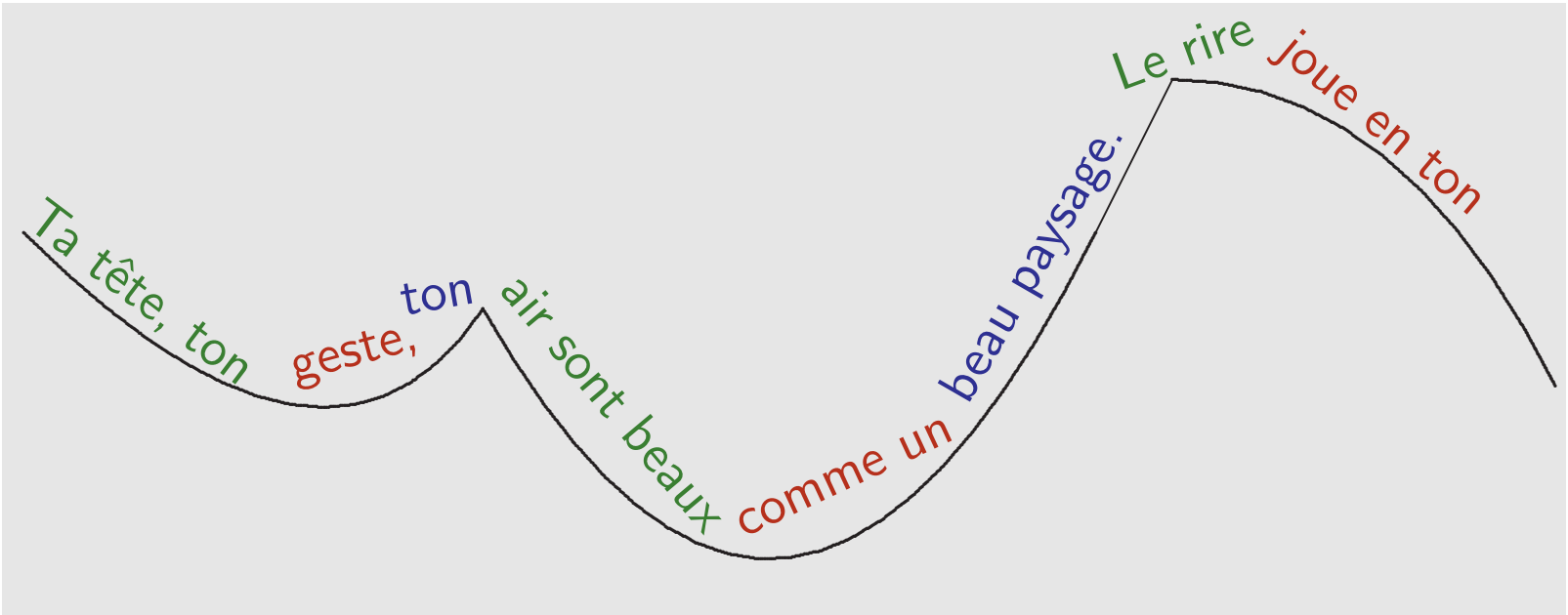



Czy przesyłana wiadomość nie zniekształciła się na wybojach?

Wiadomość jest specjalną funkcją przekształcana w krótki ciąg bitów i przekształcony skrót jest dołączany do wiadomości.

# Jednokierunkowe funkcje skrótu

## Sprawdzanie poprawności transmisji (integralności przekazu):




Czy przesyłana wiadomość nie zniekształciła się na wybojach?

Wiadomość jest specjalną funkcją przekształcana w krótki ciąg bitów i przekształcony skrót jest dołączany do wiadomości.

**Jaką funkcją można to zrobić?**

# Jednokierunkowe funkcje skrótu

***One-way hash function*** (ang. *hash* = siekanina)

# Jednokierunkowe funkcje skrótu

***One-way hash function*** (ang. *hash* = siekanina) — dla ustalonego $n$ działa tak:

| | | | |
|---|---|---|---|
| $h$ : | $\texttt{char}^*$ | $\longrightarrow$ | $\texttt{char}^n$ |
| | napisy dowolne | | napisy długości $n$ |

# Jednokierunkowe funkcje skrótu

***One-way hash function*** (ang. *hash* = siekanina) — dla ustalonego $n$ działa tak:

$$h : \underbrace{\texttt{char}^*}_{\text{napisy dowolne}} \longrightarrow \underbrace{\texttt{char}^n}_{\text{napisy długości } n}$$

przekształca napisy
**dowolne**
w napisy
**z góry ustalonej długości**

# Jednokierunkowe funkcje skrótu

***One-way hash function*** (ang. *hash* = siekanina) — dla ustalonego $n$ działa tak:

| | | | |
|---|---|---|---|
| $h:$ | $\texttt{char}^*$ | $\longrightarrow$ | $\texttt{char}^n$ |
| | napisy dowolne | | napisy długości $n$ |

przekształca napisy
**dowolne**
w napisy
**z góry ustalonej długości**

Taka funkcja **nie może** być różnowartościowa: zawsze istnieją napisy $s_1, s_2 \in \texttt{char}^*$, t.ż. $s_1 \neq s_2$ i $h(s_1) = h(s_2)$ .

# Jednokierunkowe funkcje skrótu

***One-way hash function*** (ang. *hash* = siekanina) — dla ustalonego $n$ działa tak:

| | | | |
|---|---|---|---|
| $h:$ | $\texttt{char}^*$ | $\longrightarrow$ | $\texttt{char}^n$ |
| | napisy dowolne | | napisy długości $n$ |

przekształca napisy
**dowolne**
w napisy
**z góry ustalonej długości**

Taka funkcja **nie może** być różnowartościowa: zawsze istnieją napisy $s_1, s_2 \in \texttt{char}^*$, t.ż. $s_1 \neq s_2$ i $h(s_1) = h(s_2)$ . Ale może być „obliczeniowo” różnowartościowa

# Jednokierunkowe funkcje skrótu

***One-way hash function*** (ang. *hash* = siekanina) — dla ustalonego $n$ działa tak:

$$h: \quad \underset{\text{napisy dowolne}}{\texttt{char}^*} \longrightarrow \underset{\text{napisy długości } n}{\texttt{char}^n}$$

przekształca napisy **dowolne** w napisy **z góry ustalonej długości**

Taka funkcja **nie może** być różnowartościowa: zawsze istnieją napisy $s_1, s_2 \in \texttt{char}^*$, t.ż. $s_1 \neq s_2$ i $h(s_1) = h(s_2)$ . Ale może być „obliczeniowo" różnowartościowa, czyli taka że

- obliczeniowo nierealne jest znalezienie dwóch różnych napisów $s_1, s_2 \in \texttt{char}^*$ t.ż. $h(s_1) = h(s_2)$;

# Jednokierunkowe funkcje skrótu

***One-way hash function*** (ang. *hash* = siekanina) — dla ustalonego $n$ działa tak:

$$h: \quad \underset{\text{napisy dowolne}}{\texttt{char}^*} \longrightarrow \underset{\text{napisy długości } n}{\texttt{char}^n}$$

przekształca napisy **dowolne** w napisy **z góry ustalonej długości**

Taka funkcja **nie może** być różnowartościowa: zawsze istnieją napisy $s_1, s_2 \in \texttt{char}^*$, t.ż. $s_1 \neq s_2$ i $h(s_1) = h(s_2)$. Ale może być „obliczeniowo" różnowartościowa, czyli taka że

- obliczeniowo nierealne jest znalezienie dwóch różnych napisów $s_1, s_2 \in \texttt{char}^*$ t.ż. $h(s_1) = h(s_2)$;
- dla danego napisu $t \in \texttt{char}^n$ obliczeniowo nierealne jest znalezienie takiego $s \in \texttt{char}^*$, że $h(s) = t$.

# Jednokierunkowe funkcje skrótu

## Algorytm MD5 (*Message Digest*)



Ronald Rivest
1991

# Jednokierunkowe funkcje skrótu

## Algorytm MD5 (*Message Digest*):

Przekształca $b$-bitową wiadomość $M$ ($b$ — dowolne) w jeden blok 128-bitowy.



Ronald Rivest
1991

# Jednokierunkowe funkcje skrótu

## Algorytm MD5 (*Message Digest*):

Przekształca $b$-bitową wiadomość $M$ ($b$ — dowolne) w jeden blok 128-bitowy.



Ronald Rivest
1991

**Najpierw:**
Podział $M$ na bloki 512-bitowe z 64-bitowym polem długości na końcu i dopełnieniem śmieciowymi bitami

# Jednokierunkowe funkcje skrótu

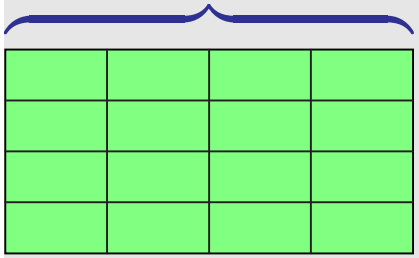
## Algorytm MD5 (*Message Digest*):

Przekształca $b$-bitową wiadomość $M$ ($b$ — dowolne) w jeden blok 128-bitowy.



Ronald Rivest
1991

**Najpierw:**
Podział $M$ na bloki 512-bitowe z 64-bitowym polem długości na końcu i dopełnieniem śmieciowymi bitami:



# Jednokierunkowe funkcje skrótu

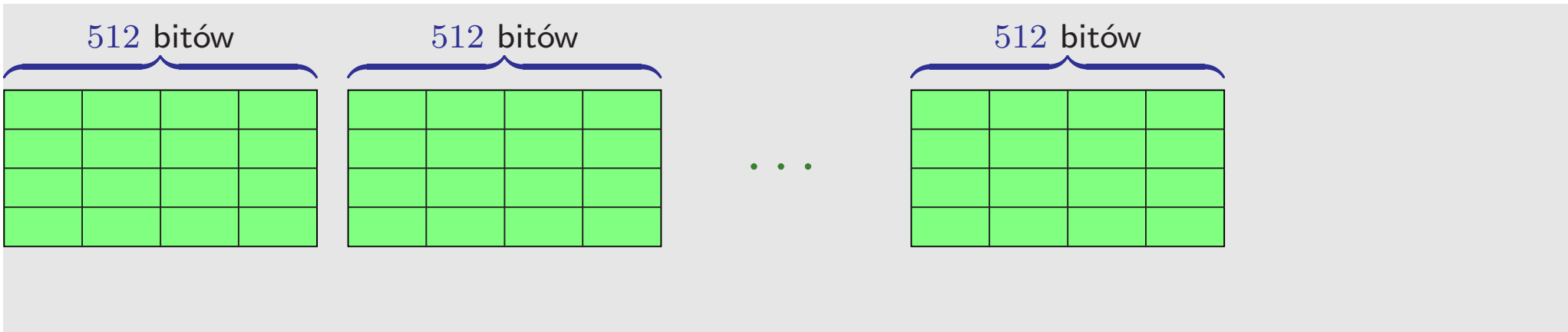
## Algorytm MD5 (*Message Digest*):

Przekształca $b$-bitową wiadomość $M$ ($b$ — dowolne) w jeden blok 128-bitowy.



Ronald Rivest
1991

**Najpierw:**
Podział $M$ na bloki 512-bitowe z 64-bitowym polem długości na końcu i dopełnieniem śmieciowymi bitami:



# Jednokierunkowe funkcje skrótu

## Algorytm MD5 (*Message Digest*):

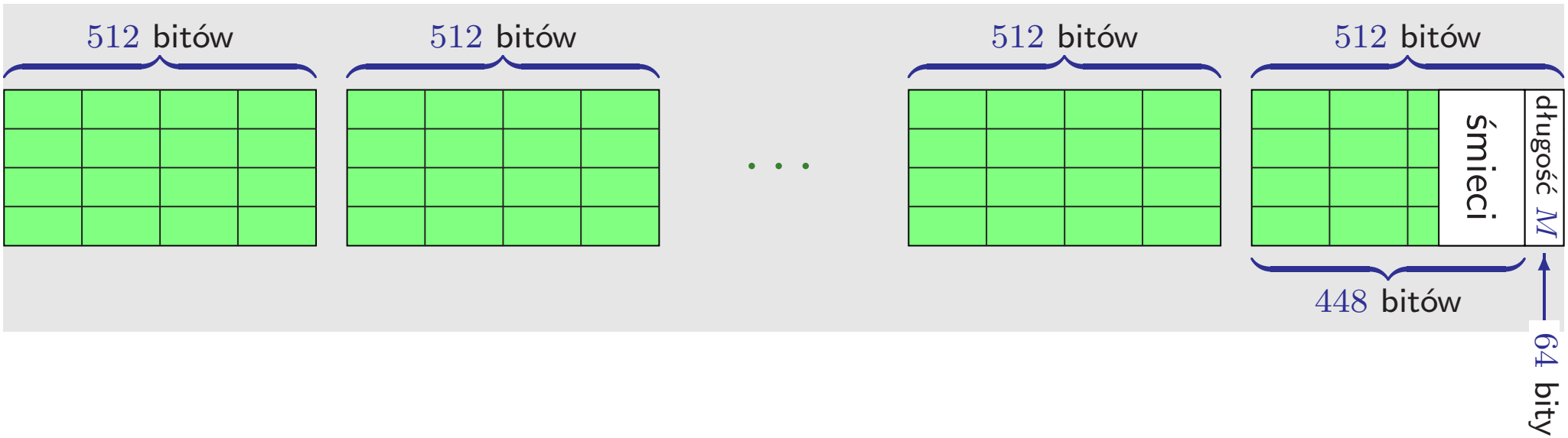
Przekształca $b$-bitową wiadomość $M$ ($b$ — dowolne) w jeden blok 128-bitowy.



Ronald Rivest
1991

## Najpierw:

Podział $M$ na bloki 512-bitowe z 64-bitowym polem długości na końcu i dopełnieniem śmieciowymi bitami:



# Jednokierunkowe funkcje skrótu

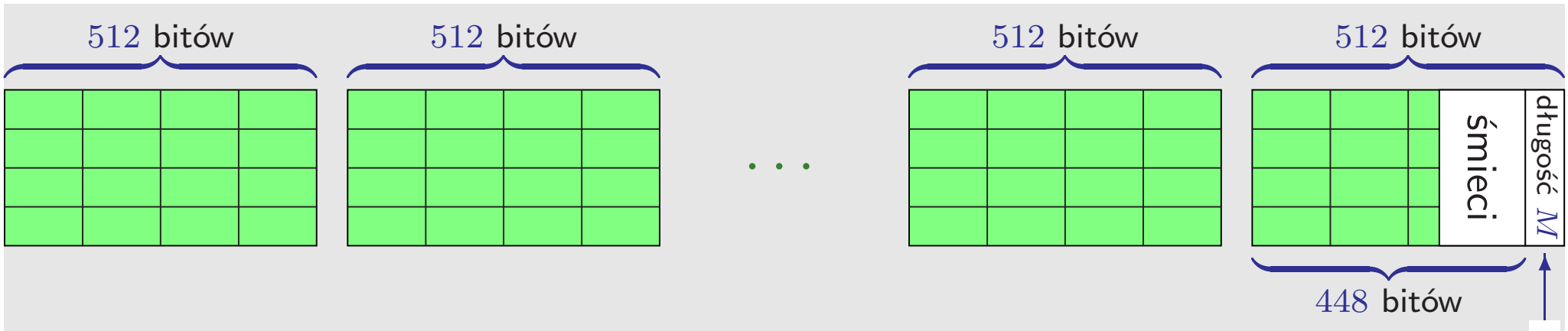
## Algorytm MD5 (*Message Digest*):

Przekształca $b$-bitową wiadomość $M$ ($b$ — dowolne) w jeden blok 128-bitowy.



Ronald Rivest
1991

**Najpierw:**
Podział $M$ na bloki 512-bitowe z 64-bitowym polem długości na końcu i dopełnieniem śmieciowymi bitami:




Każdy blok składa się z $16$ liczb $32$-bitowych.

Jak się te bloki wykorzystuje — patrz dalej.

# Jednokierunkowe funkcje skrótu

## Algorytm MD5 — ciąg dalszy

# Jednokierunkowe funkcje skrótu

**Algorytm MD5 — ciąg dalszy:**

**Następnie:**
Modyfikacja tymi blokami **stanu** złożonego z czterech zmiennych całkowitych:

| początkowo | |
|---|---|
| $A$ | 01234567 |
| $B$ | 89ABCDEF |
| $C$ | FEDCBA98 |
| $D$ | 76543210 |

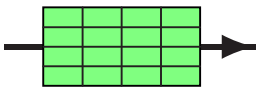
# Jednokierunkowe funkcje skrótu

**Algorytm MD5 — ciąg dalszy:**

**Następnie:**

Modyfikacja tymi blokami **stanu** złożonego z czterech zmiennych całkowitych:

| początkowo | |
|---|---|
| $A$ | 01234567 |
| $B$ | 89ABCDEF |
| $C$ | FEDCBA98 |
| $D$ | 76543210 |



| nowy stan | |
|---|---|
| $A$ | ???????? |
| $B$ | ???????? |
| $C$ | ???????? |
| $D$ | ???????? |

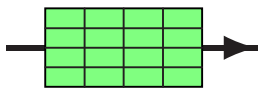
# Jednokierunkowe funkcje skrótu

**Algorytm MD5 — ciąg dalszy:**

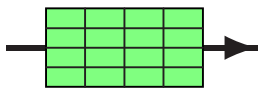
**Następnie:**

Modyfikacja tymi blokami **stanu** złożonego z czterech zmiennych całkowitych:

| początkowo | |
|---|---|
| $A$ | 01234567 |
| $B$ | 89ABCDEF |
| $C$ | FEDCBA98 |
| $D$ | 76543210 |



| nowy stan | |
|---|---|
| $A$ | ???????? |
| $B$ | ???????? |
| $C$ | ???????? |
| $D$ | ???????? |



| jeszcze nowszy stan | |
|---|---|
| $A$ | ???????? |
| $B$ | ???????? |
| $C$ | ???????? |
| $D$ | ???????? |

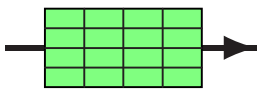
kolejne bloki wiadomości $M$

# Jednokierunkowe funkcje skrótu

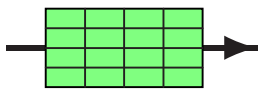
**Algorytm MD5 — ciąg dalszy:**

**Następnie:**
Modyfikacja tymi blokami **stanu** złożonego z czterech zmiennych całkowitych:

| początkowo | |
|---|---|
| $A$ | 01234567 |
| $B$ | 89ABCDEF |
| $C$ | FEDCBA98 |
| $D$ | 76543210 |



| nowy stan | |
|---|---|
| $A$ | ???????? |
| $B$ | ???????? |
| $C$ | ???????? |
| $D$ | ???????? |



| jeszcze nowszy stan | |
|---|---|
| $A$ | ???????? |
| $B$ | ???????? |
| $C$ | ???????? |
| $D$ | ???????? |

kolejne bloki wiadomości $M$

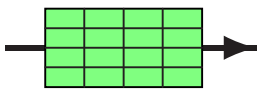
Blok ($512$-bitowy, zielony, kratkowany) przekształca stan ($128$-bitowy, ze zmiennych całkowitych) w **4 kolejnych rundach:** $R_F$, $R_G$, $R_H$, $R_I$

# Jednokierunkowe funkcje skrótu

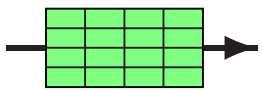
**Algorytm MD5 — ciąg dalszy:**

**Następnie:**
Modyfikacja tymi blokami **stanu** złożonego z czterech zmiennych całkowitych:

| początkowo | |
|---|---|
| $A$ | 01234567 |
| $B$ | 89ABCDEF |
| $C$ | FEDCBA98 |
| $D$ | 76543210 |



| nowy stan | |
|---|---|
| $A$ | ???????? |
| $B$ | ???????? |
| $C$ | ???????? |
| $D$ | ???????? |



| jeszcze nowszy stan | |
|---|---|
| $A$ | ???????? |
| $B$ | ???????? |
| $C$ | ???????? |
| $D$ | ???????? |

kolejne bloki wiadomości $M$

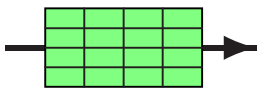
Blok ($512$-bitowy, zielony, kratkowany) przekształca stan ($128$-bitowy, ze zmiennych całkowitych) w **4 kolejnych rundach:** $R_F$, $R_G$, $R_H$, $R_I$;
w każdej używa się jednej z poniższych funkcji $\text{int} \times \text{int} \times \text{int} \longrightarrow \text{int}$

# Jednokierunkowe funkcje skrótu

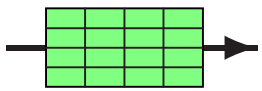
**Algorytm MD5 — ciąg dalszy:**

**Następnie:**
Modyfikacja tymi blokami **stanu** złożonego z czterech zmiennych całkowitych:

| początkowo | |
|---|---|
| $A$ | 01234567 |
| $B$ | 89ABCDEF |
| $C$ | FEDCBA98 |
| $D$ | 76543210 |



| nowy stan | |
|---|---|
| $A$ | ???????? |
| $B$ | ???????? |
| $C$ | ???????? |
| $D$ | ???????? |



| jeszcze nowszy stan | |
|---|---|
| $A$ | ???????? |
| $B$ | ???????? |
| $C$ | ???????? |
| $D$ | ???????? |

kolejne bloki wiadomości $M$

Blok ($512$-bitowy, zielony, kratkowany) przekształca stan ($128$-bitowy, ze zmiennych całkowitych) w **4 kolejnych rundach:** $R_F$, $R_G$, $R_H$, $R_I$;
w każdej używa się jednej z poniższych funkcji $\text{int} \times \text{int} \times \text{int} \longrightarrow \text{int}$ :

$$F(X,Y,Z) \stackrel{\text{def}}{=} (X \wedge Y) \vee ((\neg X) \wedge Z)$$
$$G(X,Y,Z) \stackrel{\text{def}}{=} (X \wedge Z) \vee (Y \wedge (\neg Z))$$
$$H(X,Y,Z) \stackrel{\text{def}}{=} X \oplus Y \oplus Z$$
$$I(X,Y,Z) \stackrel{\text{def}}{=} X \oplus (X \vee (\neg Z))$$

# Jednokierunkowe funkcje skrótu

**Algorytm MD5 — ciąg dalszy:**

**Następnie:**
Modyfikacja tymi blokami **stanu** złożonego z czterech zmiennych całkowitych:




Blok ($512$-bitowy, zielony, kratkowany) przekształca stan ($128$-bitowy, ze zmiennych całkowitych) w **4 kolejnych rundach:** $R_F$, $R_G$, $R_H$, $R_I$;
w każdej używa się jednej z poniższych funkcji $\text{int} \times \text{int} \times \text{int} \longrightarrow \text{int}$ :

$$F(X,Y,Z) \stackrel{\text{def}}{=} (X \wedge Y) \vee ((\neg X) \wedge Z)$$
$$G(X,Y,Z) \stackrel{\text{def}}{=} (X \wedge Z) \vee (Y \wedge (\neg Z))$$
$$H(X,Y,Z) \stackrel{\text{def}}{=} X \oplus Y \oplus Z$$
$$I(X,Y,Z) \stackrel{\text{def}}{=} X \oplus (X \vee (\neg Z))$$

**Jak** się tych funkcji używa — patrz dalej.

# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$)

# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$)**:**

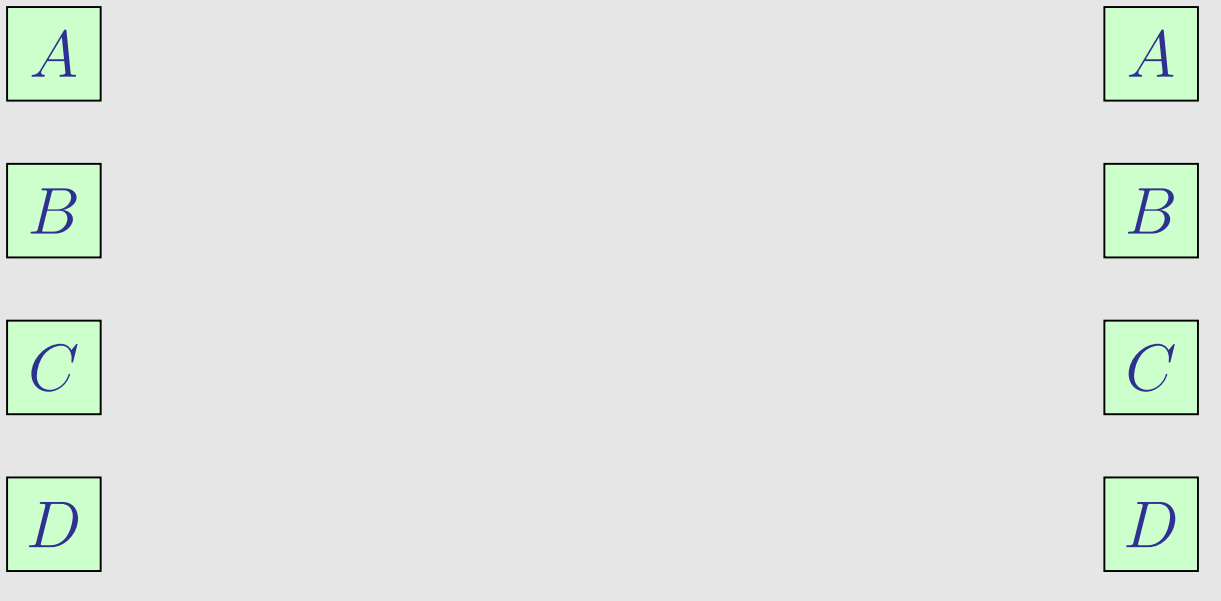
Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — $16$ liczb $32$-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$)**:**

Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — $16$ liczb $32$-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

**Krok $i$ rundy $R_k$**

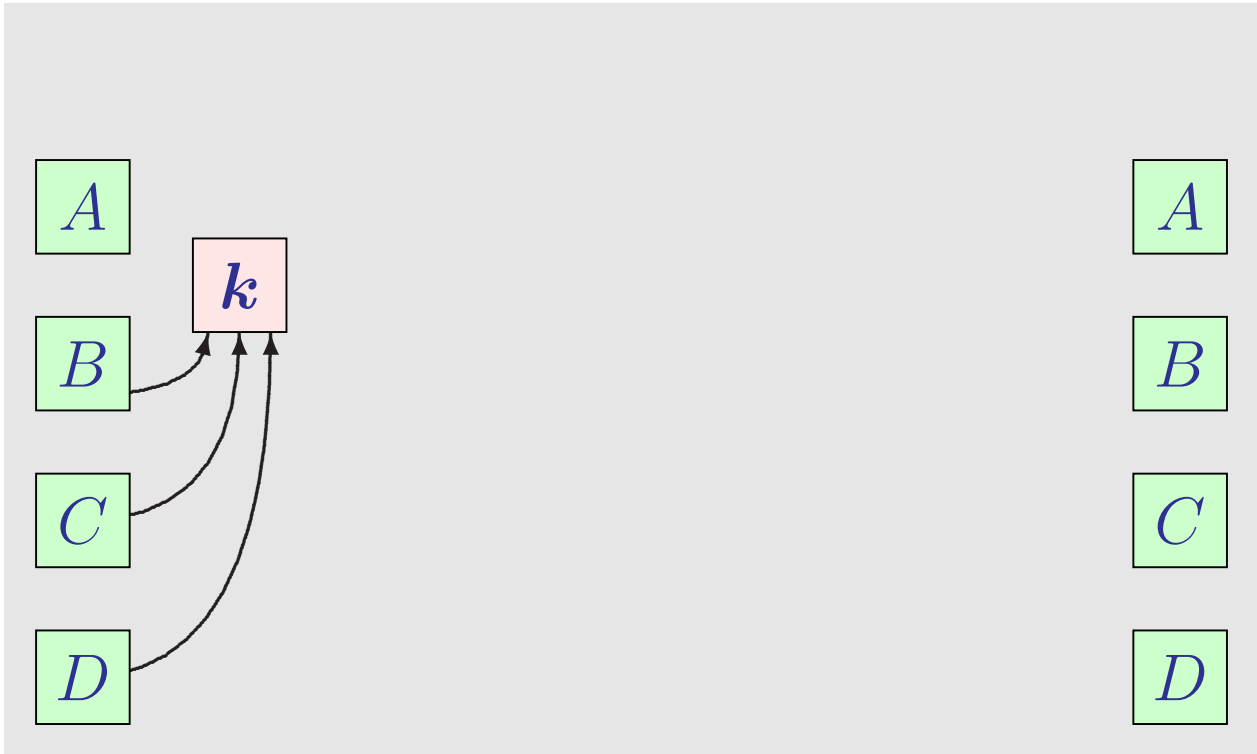


# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$):

Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — 16 liczb 32-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

**Krok $i$ rundy $R_k$**

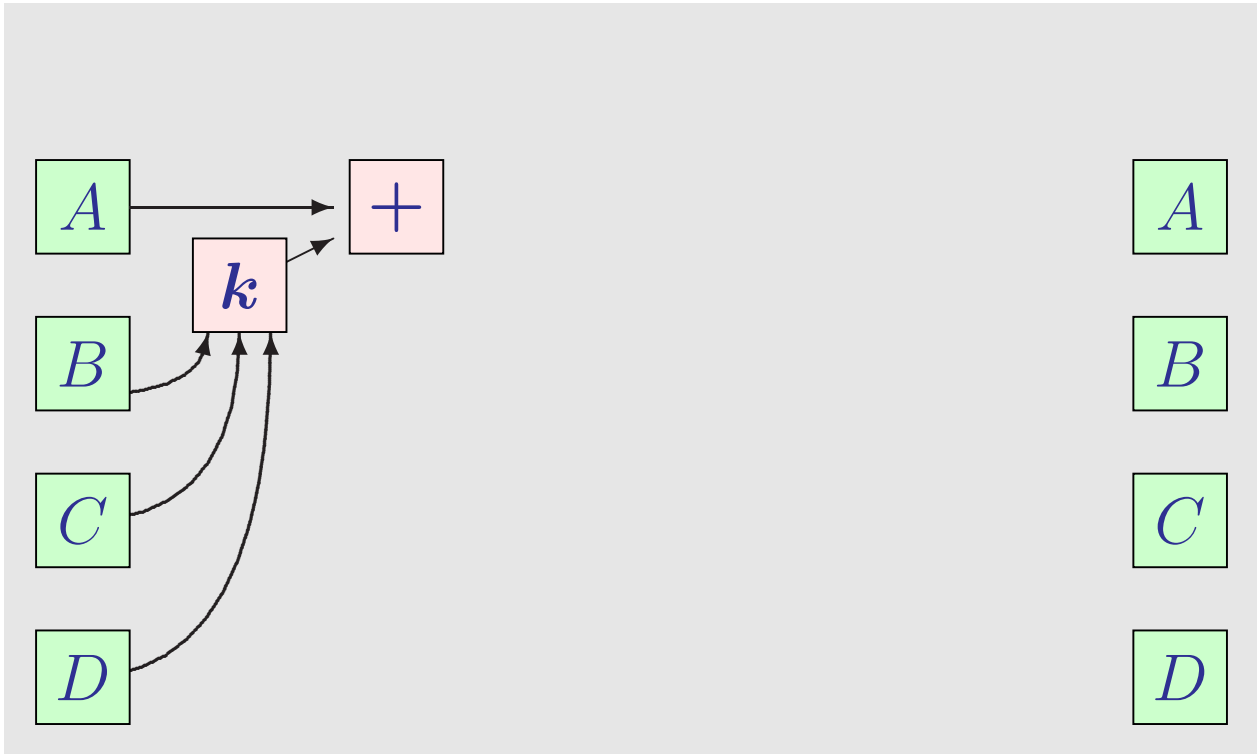


# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$):

Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — 16 liczb 32-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

**Krok $i$ rundy $R_k$**

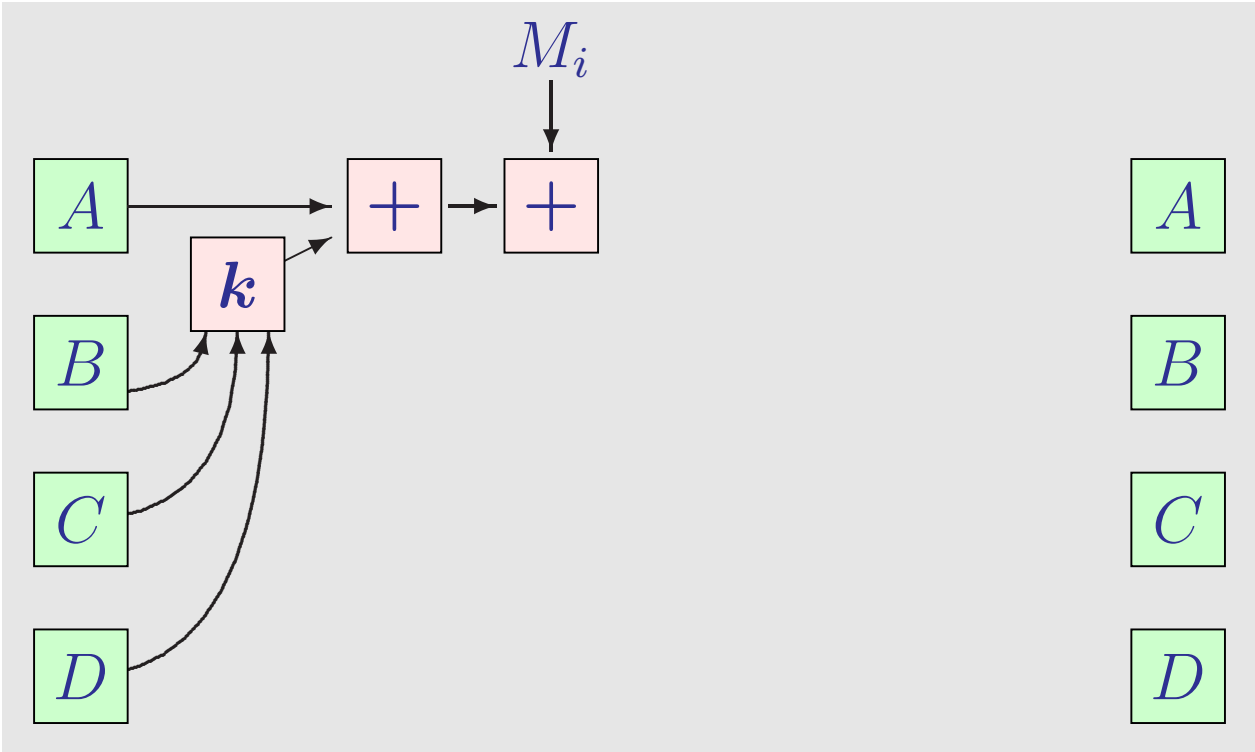


# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$)**:**

Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — 16 liczb 32-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

**Krok $i$ rundy $R_k$**

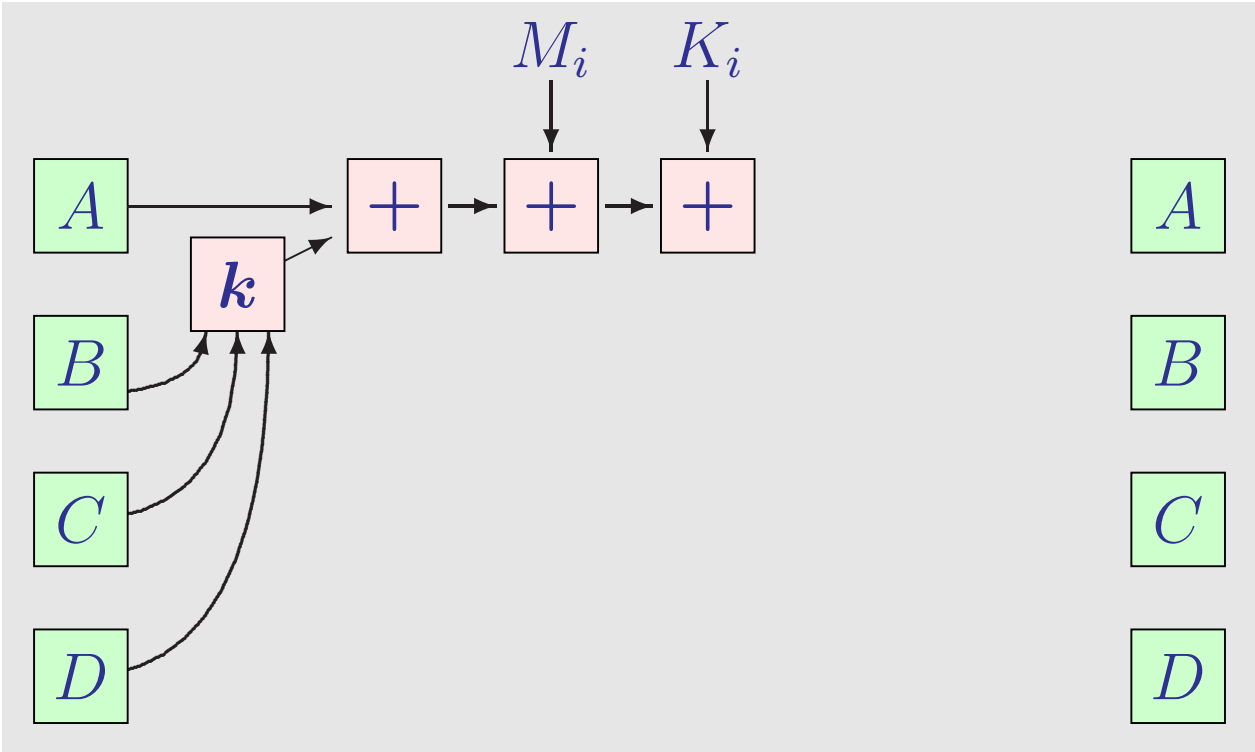


# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$)**:**

Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — 16 liczb 32-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

**Krok $i$ rundy $R_k$**

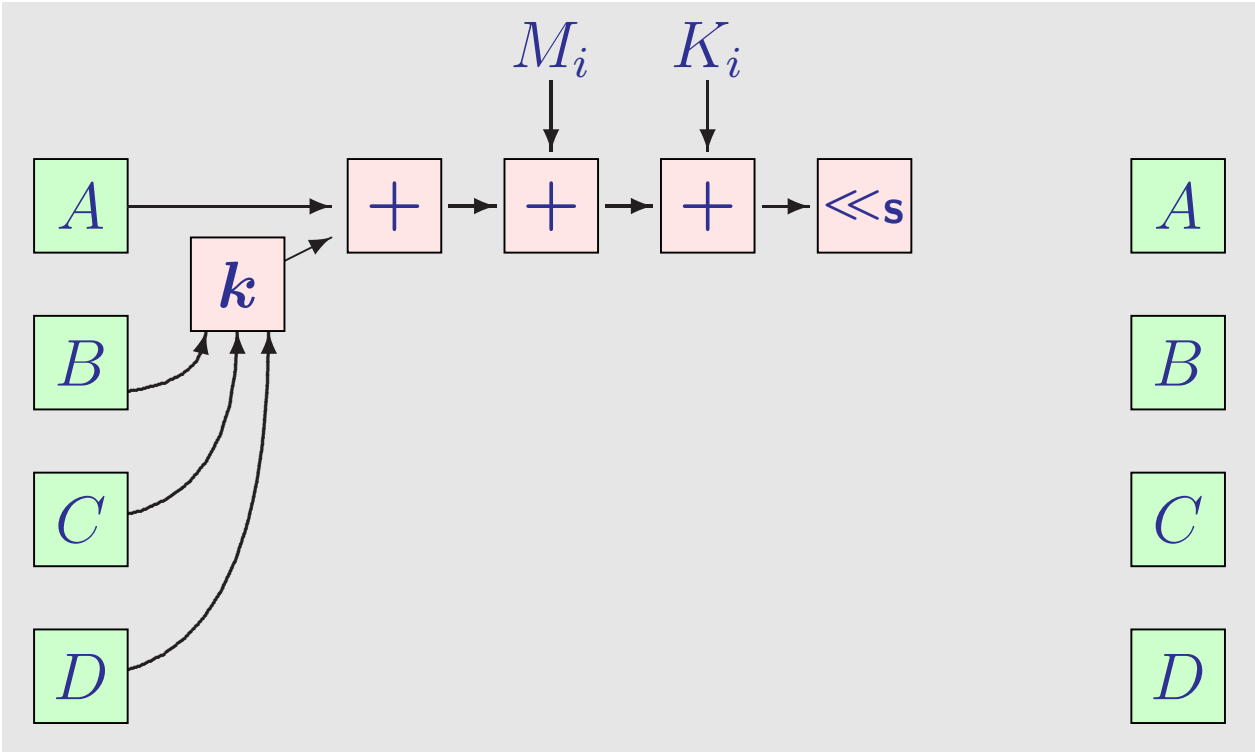


# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$)**:**

Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — 16 liczb 32-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

**Krok $i$ rundy $R_k$**

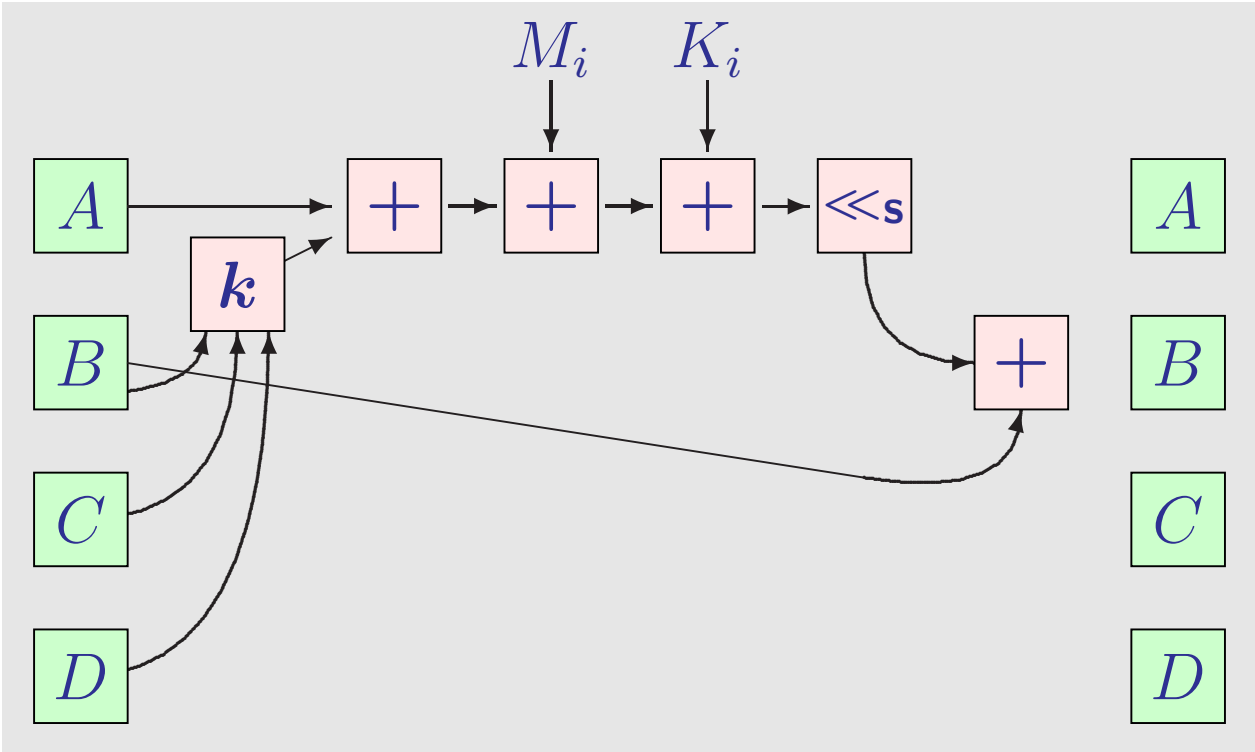


# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$):

Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — 16 liczb 32-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

**Krok $i$ rundy $R_k$**

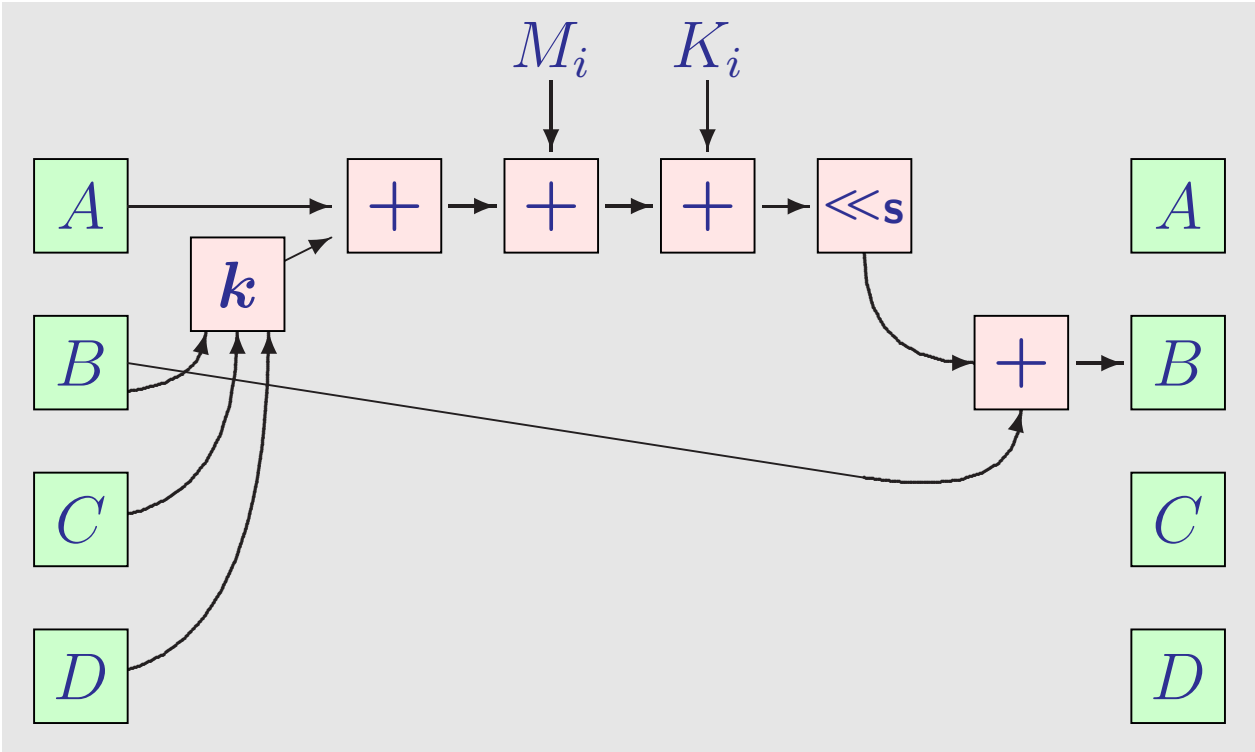


# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$):

Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — 16 liczb 32-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

**Krok $i$ rundy $R_k$**

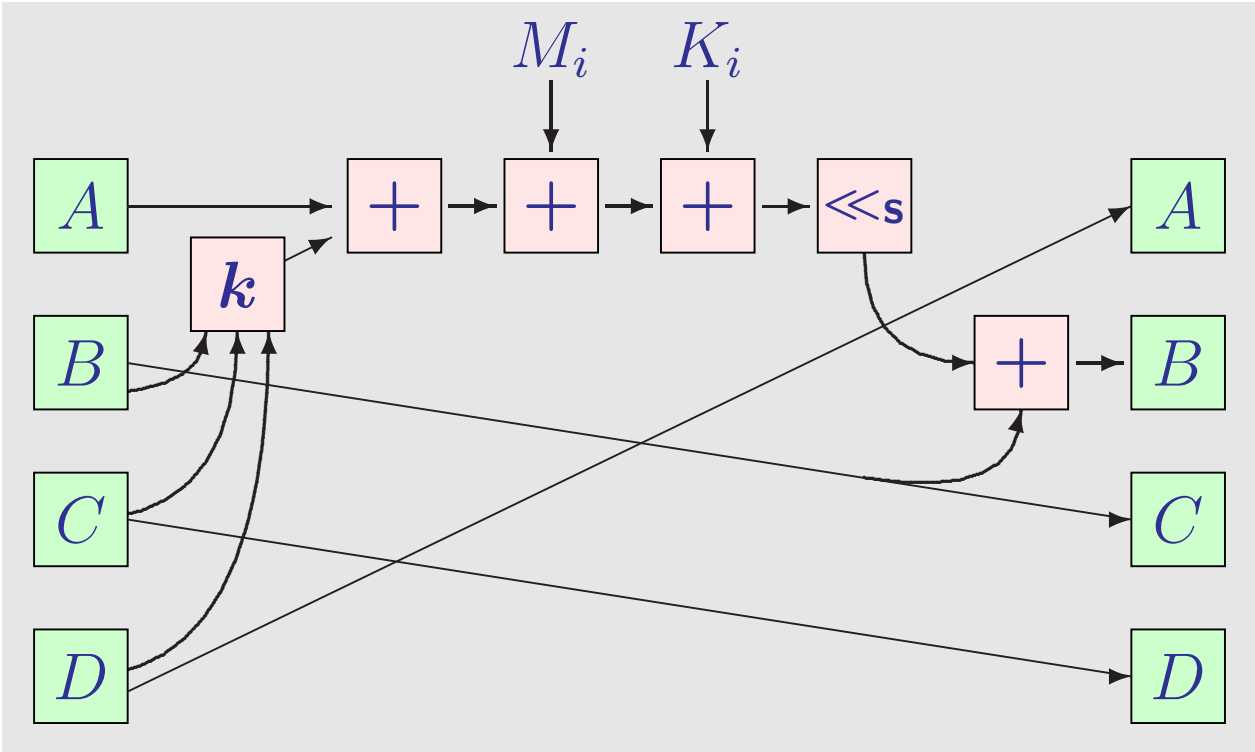


# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$)**:**

Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — 16 liczb 32-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

**Krok $i$ rundy $R_k$**



# Jednokierunkowe funkcje skrótu

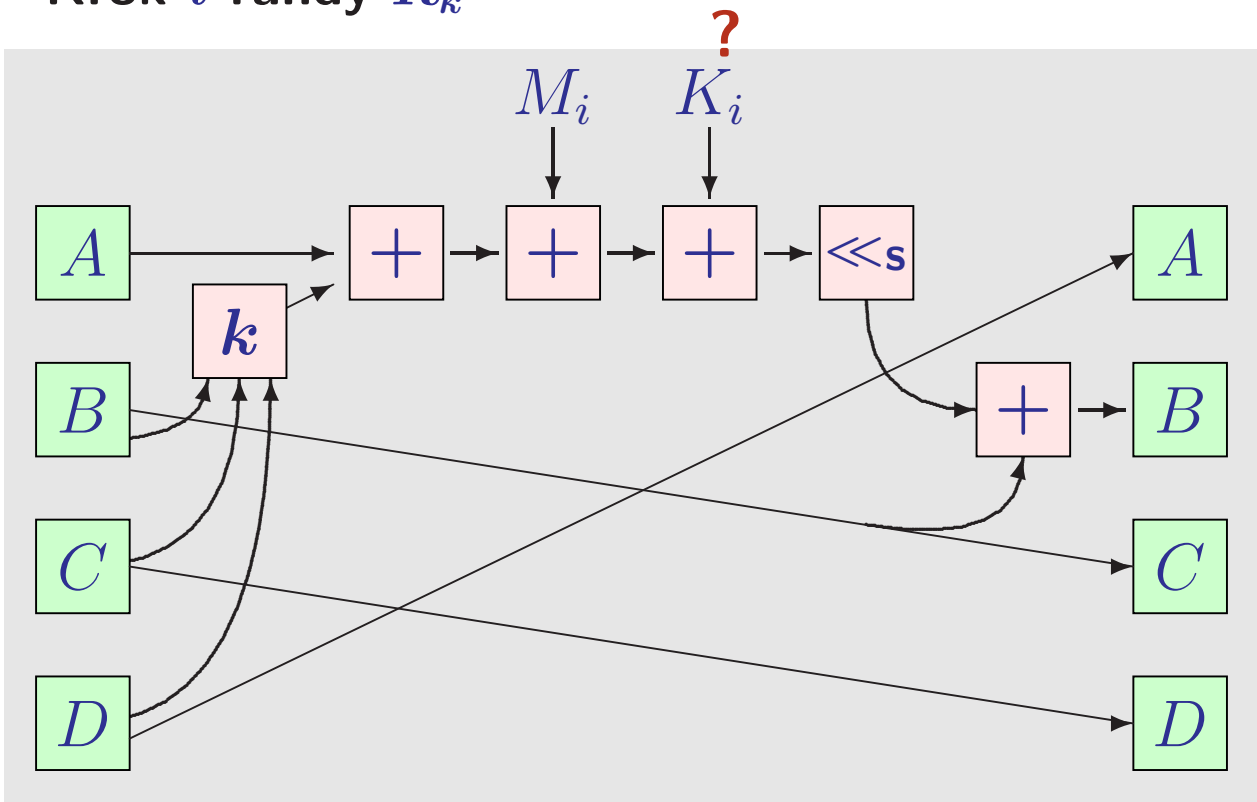
**Runda $R_k$** (dla $k \in \{F, G, H, I\}$):

Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — 16 liczb 32-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

**Na tym obrazku:**

- $K_i = \lfloor 2^{32} \cdot |\sin(j)| \rfloor$
  $j = i \cdot 16$
  $+$ numer 32-bitowej liczby w $M_i$

**Krok $i$ rundy $R_k$**

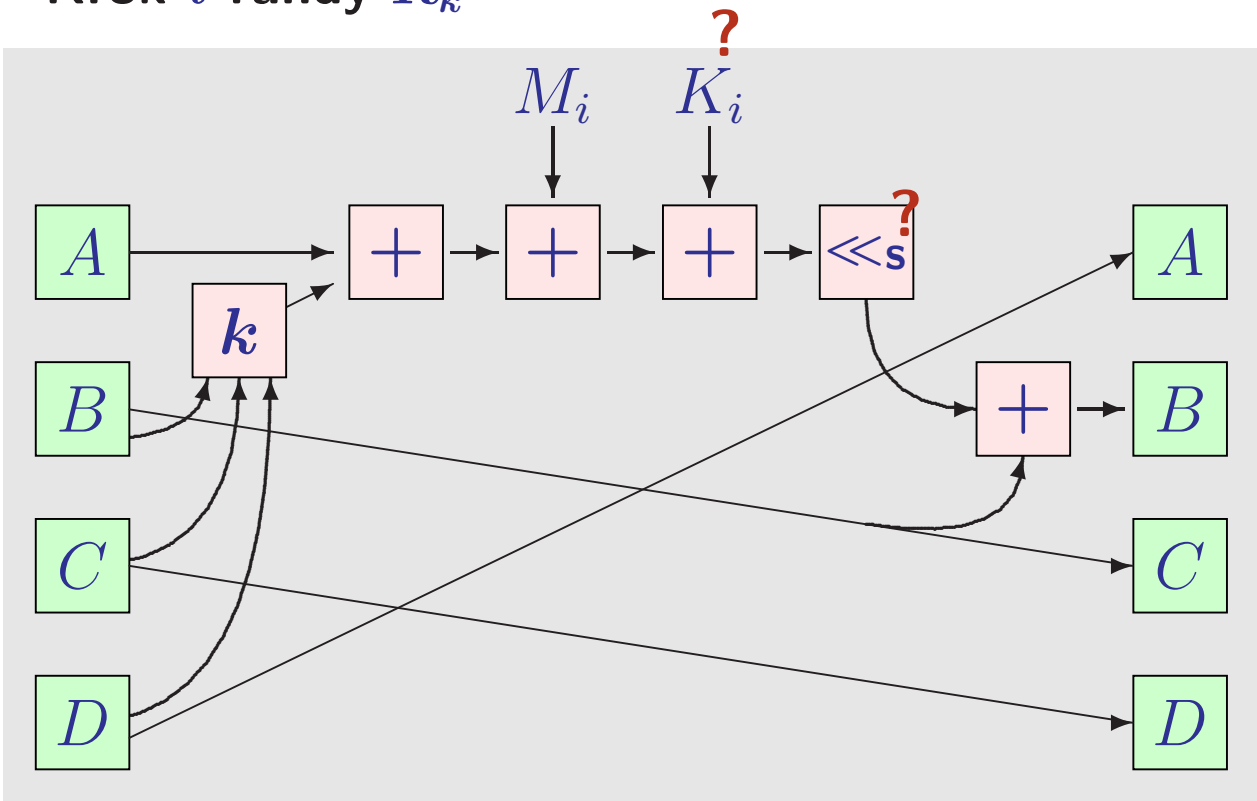


# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$):

Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — 16 liczb 32-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

**Krok $i$ rundy $R_k$**




**Na tym obrazku:**

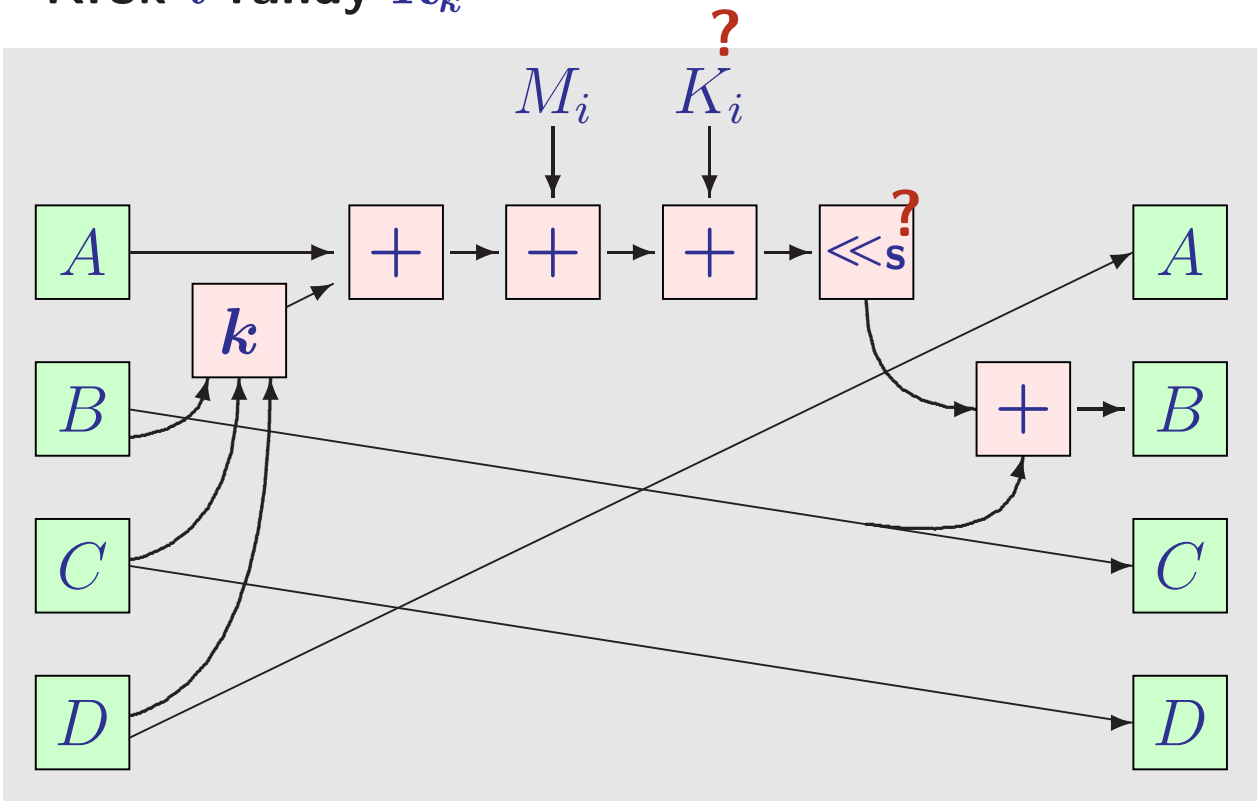
- $K_i = \lfloor 2^{32} \cdot |\sin(j)| \rfloor$
  $j = i \cdot 16$
  $+$ numer 32-bitowej liczby w $M_i$
- $<<_s$ — rotacja w lewo o $s$ bitów

# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$):

Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — 16 liczb 32-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

**Krok $i$ rundy $R_k$**




**Na tym obrazku:**

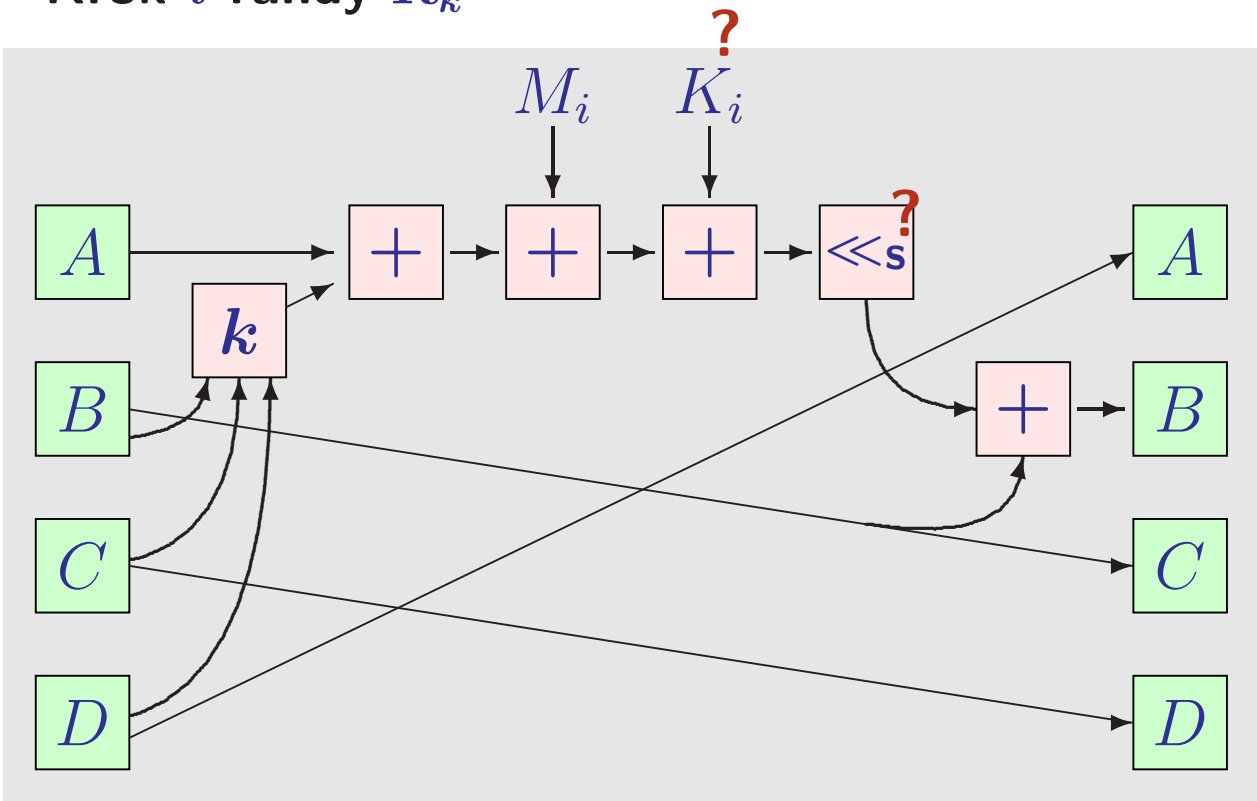
- $K_i = \lfloor 2^{32} \cdot |\sin(j)| \rfloor$
  $j = i \cdot 16$
  $+$ numer 32-bitowej liczby w $M_i$
- $<<_s$ — rotacja w lewo o $s$ bitów; $s$ jest za każdym razem inne

# Jednokierunkowe funkcje skrótu

**Runda $R_k$** (dla $k \in \{F, G, H, I\}$):

Blok wiadomości $M = (M_0, M_1, \ldots, M_{15})$ — 16 liczb 32-bitowych. Stan $ABCD$ zostaje przekształcony w 16 krokach.

**Krok $i$ rundy $R_k$**




**Na tym obrazku:**

- $K_i = \lfloor 2^{32} \cdot |\sin(j)| \rfloor$
  $j = i \cdot 16$
  $+$ numer 32-bitowej liczby w $M_i$
- $<<_s$ — rotacja w lewo o $s$ bitów; $s$ jest za każdym razem inne; tablicę jego wartości można znaleźć np. w https://en.wikipedia.org/wiki/MD5#Algorithm

# Jednokierunkowe funkcje skrótu

Szyfrowanie jest skomplikowane, a teraz

**JAK TO ODSZYFROWAĆ?**

# Jednokierunkowe funkcje skrótu

Szyfrowanie jest skomplikowane, a teraz

## JAK TO ODSZYFROWAĆ?

**Odp. 1:** Mamy nadzieję, że **nie da się**; że szyfrowanie jest nieodwracalne.

# Jednokierunkowe funkcje skrótu

Szyfrowanie jest skomplikowane, a teraz

## JAK TO ODSZYFROWAĆ?

**Odp. 1:** Mamy nadzieję, że **nie da się**; że szyfrowanie jest nieodwracalne.

**Odp. 2:** Odszyfrowywania **nie potrzebujemy**.

# Jednokierunkowe funkcje skrótu

Szyfrowanie jest skomplikowane, a teraz

## JAK TO ODSZYFROWAĆ?

**Odp. 1:** Mamy nadzieję, że **nie da się**; że szyfrowanie jest nieodwracalne.

**Odp. 2:** Odszyfrowywania **nie potrzebujemy**.

- **Hasła dostępu** —

# Jednokierunkowe funkcje skrótu

Szyfrowanie jest skomplikowane, a teraz

## JAK TO ODSZYFROWAĆ?

**Odp. 1:** Mamy nadzieję, że **nie da się**; że szyfrowanie jest nieodwracalne.

**Odp. 2:** Odszyfrowywania **nie potrzebujemy**.

- **Hasła dostępu** — zapamiętywany na serwerze jest skrót — wynik szyfrowania.

# Jednokierunkowe funkcje skrótu

Szyfrowanie jest skomplikowane, a teraz

## JAK TO ODSZYFROWAĆ?

**Odp. 1:** Mamy nadzieję, że **nie da się**; że szyfrowanie jest nieodwracalne.

**Odp. 2:** Odszyfrowywania **nie potrzebujemy**.

- **Hasła dostępu** — zapamiętywany na serwerze jest skrót — wynik szyfrowania. Kiedy użytkownik się loguje, podane przez niego hasło jest za każdym razem na nowo szyfrowane

# Jednokierunkowe funkcje skrótu

Szyfrowanie jest skomplikowane, a teraz

## JAK TO ODSZYFROWAĆ?

**Odp. 1:** Mamy nadzieję, że **nie da się**; że szyfrowanie jest nieodwracalne.

**Odp. 2:** Odszyfrowywania **nie potrzebujemy**.

- **Hasła dostępu** — zapamiętywany na serwerze jest skrót — wynik szyfrowania. Kiedy użytkownik się loguje, podane przez niego hasło jest za każdym razem na nowo szyfrowane i porównywane bit po bicie z zapamiętanym skrótem.

# Jednokierunkowe funkcje skrótu

Szyfrowanie jest skomplikowane, a teraz

## JAK TO ODSZYFROWAĆ?

**Odp. 1:** Mamy nadzieję, że **nie da się**; że szyfrowanie jest nieodwracalne.

**Odp. 2:** Odszyfrowywania **nie potrzebujemy**.

- **Hasła dostępu** — zapamiętywany na serwerze jest skrót — wynik szyfrowania. Kiedy użytkownik się loguje, podane przez niego hasło jest za każdym razem na nowo szyfrowane i porównywane bit po bicie z zapamiętanym skrótem.
- **Integralność** —

# Jednokierunkowe funkcje skrótu

Szyfrowanie jest skomplikowane, a teraz

## JAK TO ODSZYFROWAĆ?

**Odp. 1:** Mamy nadzieję, że **nie da się**; że szyfrowanie jest nieodwracalne.

**Odp. 2:** Odszyfrowywania **nie potrzebujemy**.

- **Hasła dostępu** — zapamiętywany na serwerze jest skrót — wynik szyfrowania. Kiedy użytkownik się loguje, podane przez niego hasło jest za każdym razem na nowo szyfrowane i porównywane bit po bicie z zapamiętanym skrótem.
- **Integralność** — do wiadomości dołączany jest jej skrót — wynik szyfrowania.

# Jednokierunkowe funkcje skrótu

Szyfrowanie jest skomplikowane, a teraz

## JAK TO ODSZYFROWAĆ?

**Odp. 1:** Mamy nadzieję, że **nie da się**; że szyfrowanie jest nieodwracalne.

**Odp. 2:** Odszyfrowywania **nie potrzebujemy**.

- **Hasła dostępu** — zapamiętywany na serwerze jest skrót — wynik szyfrowania. Kiedy użytkownik się loguje, podane przez niego hasło jest za każdym razem na nowo szyfrowane i porównywane bit po bicie z zapamiętanym skrótem.
- **Integralność** — do wiadomości dołączany jest jej skrót — wynik szyfrowania. Odbiorca szyfruje otrzymaną wiadomość i porównuje bit po bicie z dołączonym skrótem.

# Jednokierunkowe funkcje skrótu

## Wątpliwości na temat MD5

# Jednokierunkowe funkcje skrótu

**Wątpliwości na temat MD5:**

- Czy jest możliwe, żeby dwaj użytkownicy mieli różne hasła, ale z identycznymi skrótami?

# Jednokierunkowe funkcje skrótu

**Wątpliwości na temat MD5:**

- Czy jest możliwe, żeby dwaj użytkownicy mieli różne hasła, ale z identycznymi skrótami?
  **Odp.:** Formalnie **tak**, bo nie ma dowodu, że algorytm jest obliczeniowo różnowartościowy. Praktycznie — **nie**.

# Jednokierunkowe funkcje skrótu

**Wątpliwości na temat MD5:**

- Czy jest możliwe, żeby dwaj użytkownicy mieli różne hasła, ale z identycznymi skrótami?
  **Odp.:** Formalnie **tak**, bo nie ma dowodu, że algorytm jest obliczeniowo różnowartościowy. Praktycznie — **nie**.

- Czy **na pewno** znając skrót nie da się wyliczyć hasła?

# Jednokierunkowe funkcje skrótu

**Wątpliwości na temat MD5:**

- Czy jest możliwe, żeby dwaj użytkownicy mieli różne hasła, ale z identycznymi skrótami?
  **Odp.:** Formalnie **tak**, bo nie ma dowodu, że algorytm jest obliczeniowo różnowartościowy. Praktycznie — **nie**.

- Czy **na pewno** znając skrót nie da się wyliczyć hasła?
  **Odp.:** Jest to trudne, ale od 2008 wiadomo, że MD5 da się złamać.

# Jednokierunkowe funkcje skrótu

**Wątpliwości na temat MD5:**

- Czy jest możliwe, żeby dwaj użytkownicy mieli różne hasła, ale z identycznymi skrótami?
  **Odp.:** Formalnie **tak**, bo nie ma dowodu, że algorytm jest obliczeniowo różnowartościowy. Praktycznie — **nie**.

- Czy **na pewno** znając skrót nie da się wyliczyć hasła?
  **Odp.:** Jest to trudne, ale od 2008 wiadomo, że MD5 da się złamać.
  Obecnie MD5
  - nie jest używany do przechowywania haseł

# Jednokierunkowe funkcje skrótu

**Wątpliwości na temat MD5:**

- Czy jest możliwe, żeby dwaj użytkownicy mieli różne hasła, ale z identycznymi skrótami?
  **Odp.:** Formalnie **tak**, bo nie ma dowodu, że algorytm jest obliczeniowo różnowartościowy. Praktycznie — **nie**.

- Czy **na pewno** znając skrót nie da się wyliczyć hasła?
  **Odp.:** Jest to trudne, ale od 2008 wiadomo, że MD5 da się złamać.
  Obecnie MD5
  - nie jest używany do przechowywania haseł,
  - jest używany do kontroli integralności.

# Jednokierunkowe funkcje skrótu

**Wątpliwości na temat MD5:**

- Czy jest możliwe, żeby dwaj użytkownicy mieli różne hasła, ale z identycznymi skrótami?
  **Odp.:** Formalnie **tak**, bo nie ma dowodu, że algorytm jest obliczeniowo różnowartościowy. Praktycznie — **nie**.

- Czy **na pewno** znając skrót nie da się wyliczyć hasła?
  **Odp.:** Jest to trudne, ale od 2008 wiadomo, że MD5 da się złamać.
  Obecnie MD5
  - nie jest używany do przechowywania haseł,
  - jest używany do kontroli integralności.

- Co zamiast MD5?

# Jednokierunkowe funkcje skrótu

**Wątpliwości na temat MD5:**

- Czy jest możliwe, żeby dwaj użytkownicy mieli różne hasła, ale z identycznymi skrótami?
  **Odp.:** Formalnie **tak**, bo nie ma dowodu, że algorytm jest obliczeniowo różnowartościowy. Praktycznie — **nie**.

- Czy **na pewno** znając skrót nie da się wyliczyć hasła?
  **Odp.:** Jest to trudne, ale od 2008 wiadomo, że MD5 da się złamać.
  Obecnie MD5
  - nie jest używany do przechowywania haseł,
  - jest używany do kontroli integralności.

- Co zamiast MD5?
  **Odp.:**

| **algorytm** | **pełna nazwa** | **od kiedy** | **wielkość skrótu** |
|---|---|---|---|
| SHA-1 | *Secure* | 1995 | 160 Bitów |
| SHA-2 | *Hash* | 2001 | od 256 do 512 bitów |
| SHA-3 | *Algorithm* | 2015 | różnie |

# Jednokierunkowe funkcje skrótu

**Wątpliwości na temat MD5:**

- Czy jest możliwe, żeby dwaj użytkownicy mieli różne hasła, ale z identycznymi skrótami?
  **Odp.:** Formalnie **tak**, bo nie ma dowodu, że algorytm jest obliczeniowo różnowartościowy. Praktycznie — **nie**.

- Czy **na pewno** znając skrót nie da się wyliczyć hasła?
  **Odp.:** Jest to trudne, ale od 2008 wiadomo, że MD5 da się złamać.
  Obecnie MD5
  - nie jest używany do przechowywania haseł,
  - jest używany do kontroli integralności.

- Co zamiast MD5?
  **Odp.:**

| **algorytm** | **pełna nazwa** | **od kiedy** | **wielkość skrótu** |
|---|---|---|---|
| SHA-1 | *Secure* | 1995 | 160 Bitów |
| SHA-2 | *Hash* | 2001 | od 256 do 512 bitów |
| SHA-3 | *Algorithm* | 2015 | różnie |
| BLAKE | | 2012 | od 224 do 512 bitów |

# Jednokierunkowe funkcje skrótu

**Wątpliwości na temat MD5:**

- Czy jest możliwe, żeby dwaj użytkownicy mieli różne hasła, ale z identycznymi skrótami?
  **Odp.:** Formalnie **tak**, bo nie ma dowodu, że algorytm jest obliczeniowo różnowartościowy. Praktycznie — **nie**.

- Czy **na pewno** znając skrót nie da się wyliczyć hasła?
  **Odp.:** Jest to trudne, ale od 2008 wiadomo, że MD5 da się złamać.
  Obecnie MD5
  - nie jest używany do przechowywania haseł,
  - jest używany do kontroli integralności.

- Co zamiast MD5?
  **Odp.:**

| **algorytm** | **pełna nazwa** | **od kiedy** | **wielkość skrótu** |
|---|---|---|---|
| SHA-1 | *Secure* | 1995 | 160 Bitów |
| SHA-2 | *Hash* | 2001 | od 256 do 512 bitów |
| SHA-3 | *Algorithm* | 2015 | różnie |
| BLAKE | | 2012 | od 224 do 512 bitów |

. . . jest jeszcze wiele innych...

# Kryptografia krzywych eliptycznych

# Kryptografia krzywych eliptycznych

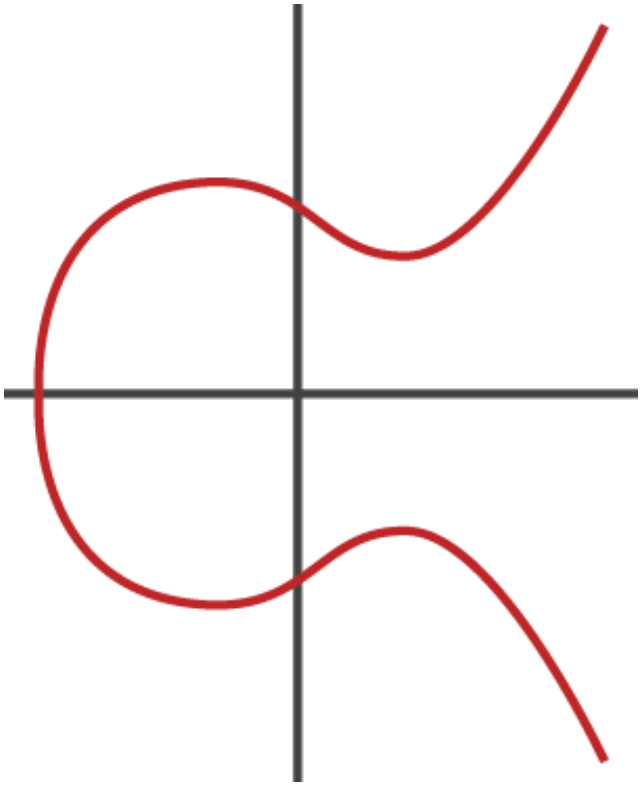
## Krzywa eliptyczna:

$$y^2 = x^3 + a \cdot x + b$$

# Kryptografia krzywych eliptycznych

## Krzywa eliptyczna:

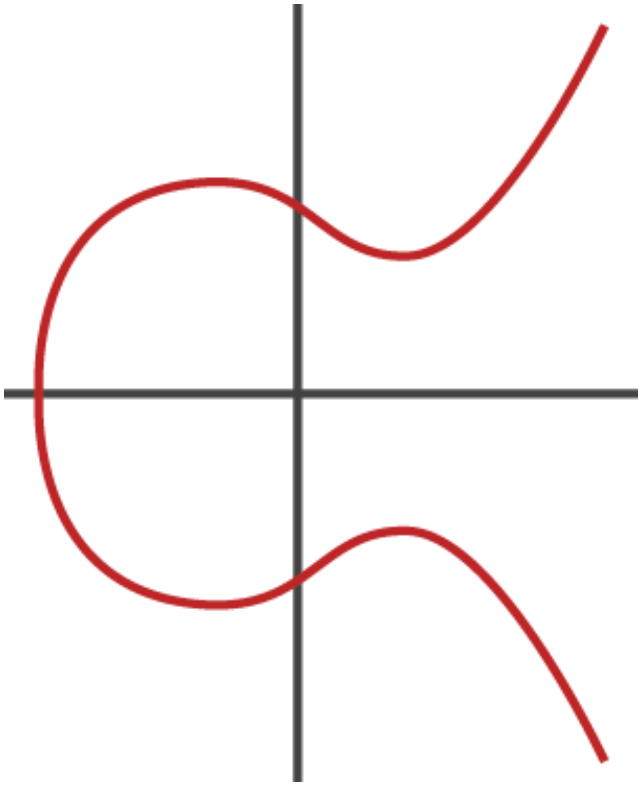
$$y^2 = x^3 + a \cdot x + b$$



# Kryptografia krzywych eliptycznych

## Krzywa eliptyczna:

$$y^2 = x^3 + a \cdot x + b$$

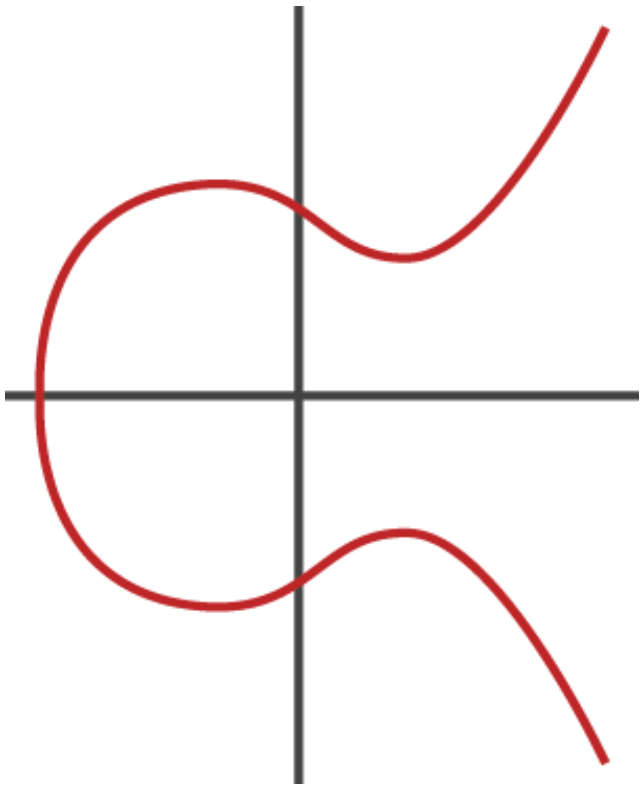


- przecina oś $y$ w pktach $\sqrt{b}$ i $-\sqrt{b}$;

# Kryptografia krzywych eliptycznych

## Krzywa eliptyczna:

$$y^2 = x^3 + a \cdot x + b$$

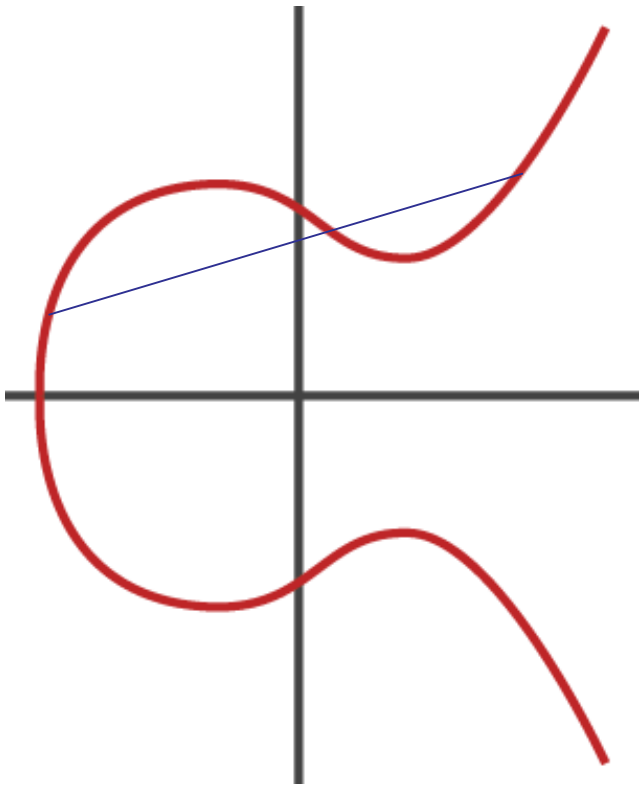


- przecina oś $y$ w pktach $\sqrt{b}$ i $-\sqrt{b}$;
- symetryczna względem osi $x$

# Kryptografia krzywych eliptycznych

## Krzywa eliptyczna:

$$y^2 = x^3 + a \cdot x + b$$

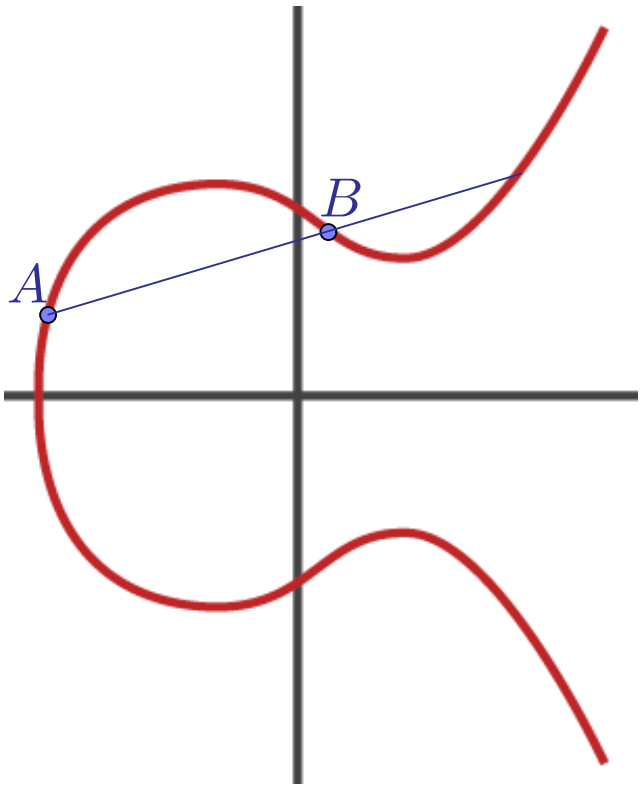


- przecina oś $y$ w pktach $\sqrt{b}$ i $-\sqrt{b}$;
- symetryczna względem osi $x$;
- dowolną prostą przecina w $\leqslant 3$ punktach

# Kryptografia krzywych eliptycznych

## Krzywa eliptyczna:

$$y^2 = x^3 + a \cdot x + b$$

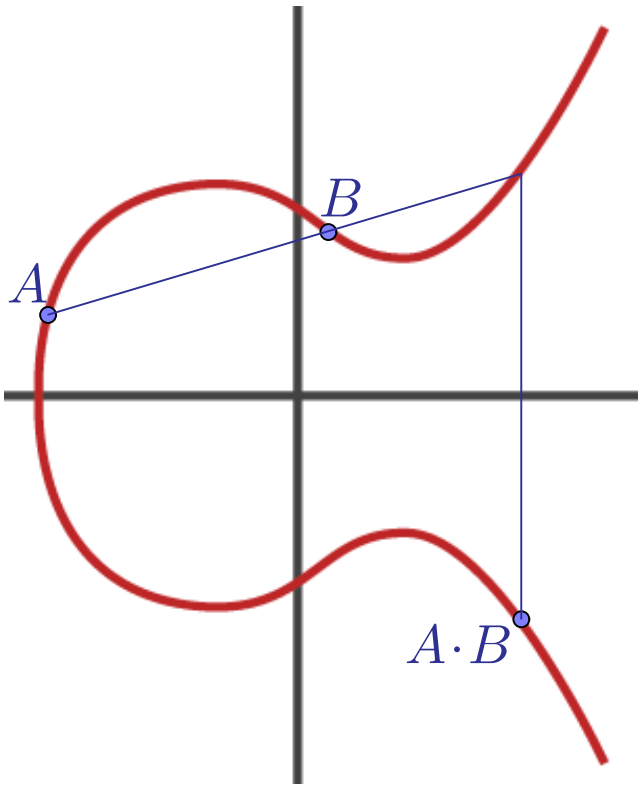



- przecina oś $y$ w pktach $\sqrt{b}$ i $-\sqrt{b}$;
- symetryczna względem osi $x$;
- dowolną prostą przecina w $\leqslant 3$ punktach;
- niech $A$ i $B$ leżą na krzywej w taki sposób, że ma ona jeszcze jeden punkt przecięcia z prostą $AB$

# Kryptografia krzywych eliptycznych

## Krzywa eliptyczna:

$$y^2 = x^3 + a \cdot x + b$$

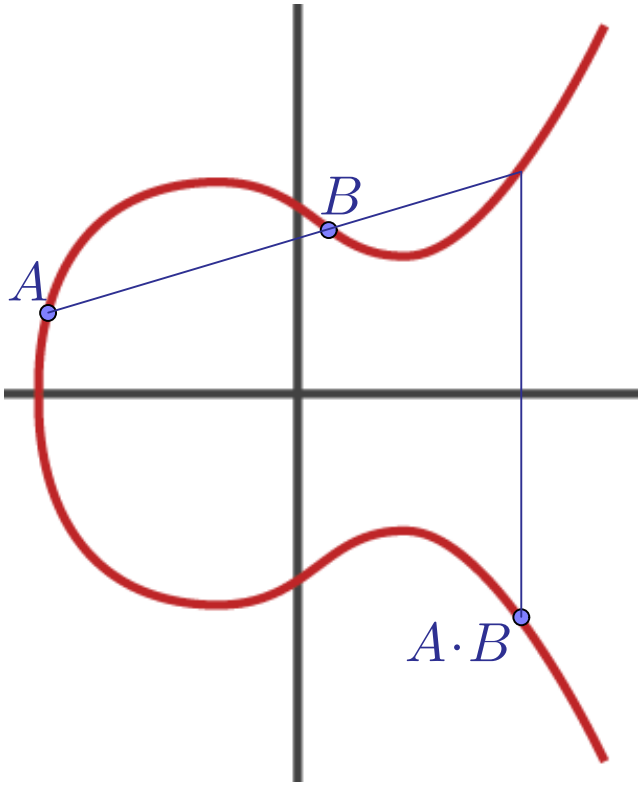



- przecina oś $y$ w pktach $\sqrt{b}$ i $-\sqrt{b}$;
- symetryczna względem osi $x$;
- dowolną prostą przecina w $\leqslant 3$ punktach;
- niech $A$ i $B$ leżą na krzywej w taki sposób, że ma ona jeszcze jeden punkt przecięcia z prostą $AB$; przez $A \cdot B$ rozumiemy punkt krzywej symetryczny do niego wzgl. osi $x$

# Kryptografia krzywych eliptycznych

## Krzywa eliptyczna:

$$y^2 = x^3 + a \cdot x + b$$

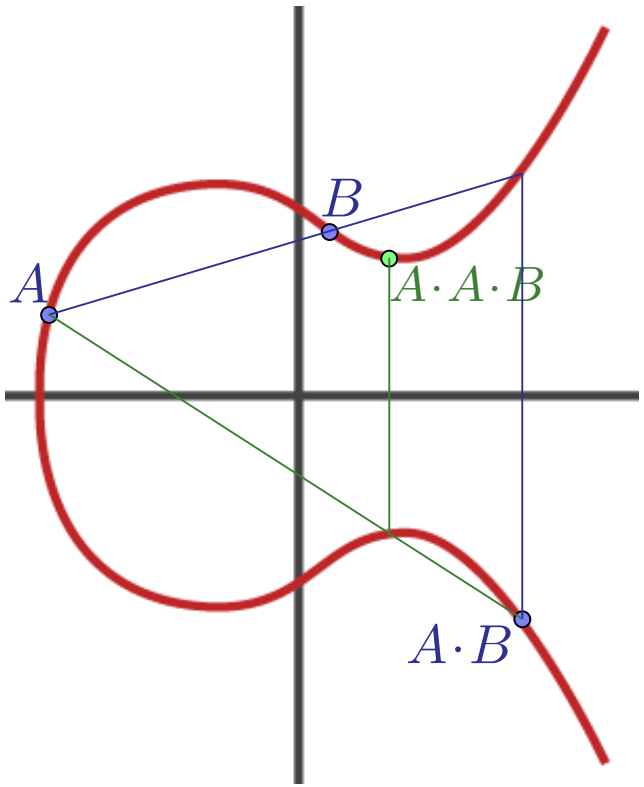



- przecina oś $y$ w pktach $\sqrt{b}$ i $-\sqrt{b}$;
- symetryczna względem osi $x$;
- dowolną prostą przecina w $\leqslant 3$ punktach;
- niech $A$ i $B$ leżą na krzywej w taki sposób, że ma ona jeszcze jeden punkt przecięcia z prostą $AB$; przez $A \cdot B$ rozumiemy punkt krzywej symetryczny do niego wzgl. osi $x$;
- operację $\cdot$ można stosować wielokrotnie: $A \cdot B$

# Kryptografia krzywych eliptycznych

## Krzywa eliptyczna:

$$y^2 = x^3 + a \cdot x + b$$

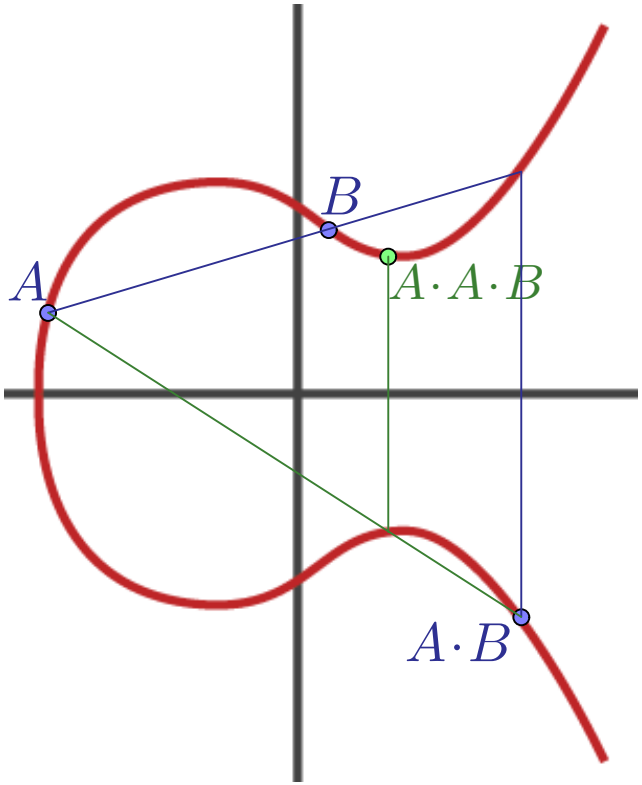



- przecina oś $y$ w pktach $\sqrt{b}$ i $-\sqrt{b}$;
- symetryczna względem osi $x$;
- dowolną prostą przecina w $\leqslant 3$ punktach;
- niech $A$ i $B$ leżą na krzywej w taki sposób, że ma ona jeszcze jeden punkt przecięcia z prostą $AB$; przez $A \cdot B$ rozumiemy punkt krzywej symetryczny do niego wzgl. osi $x$;
- operację $\cdot$ można stosować wielokrotnie:
  $A \cdot B$
  $A \cdot A \cdot B$

# Kryptografia krzywych eliptycznych

## Krzywa eliptyczna:

$$y^2 = x^3 + a \cdot x + b$$

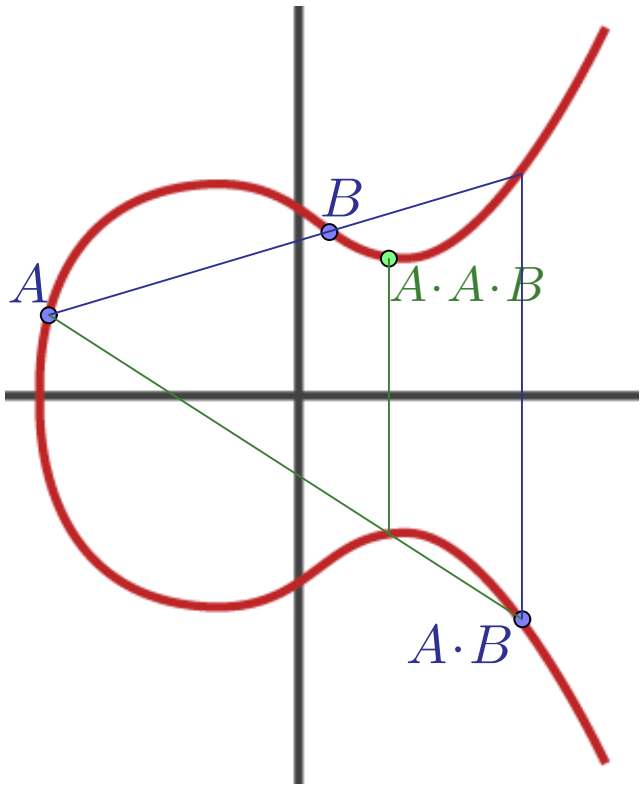



- przecina oś $y$ w pktach $\sqrt{b}$ i $-\sqrt{b}$;
- symetryczna względem osi $x$;
- dowolną prostą przecina w $\leqslant 3$ punktach;
- niech $A$ i $B$ leżą na krzywej w taki sposób, że ma ona jeszcze jeden punkt przecięcia z prostą $AB$; przez $A \cdot B$ rozumiemy punkt krzywej symetryczny do niego wzgl. osi $x$;
- operację $\cdot$ można stosować wielokrotnie:
  $A \cdot B$
  $A \cdot A \cdot B$
  $A \cdot A \cdot A \cdot B$
  ...

# Kryptografia krzywych eliptycznych

## Krzywa eliptyczna:

$$y^2 = x^3 + a \cdot x + b$$



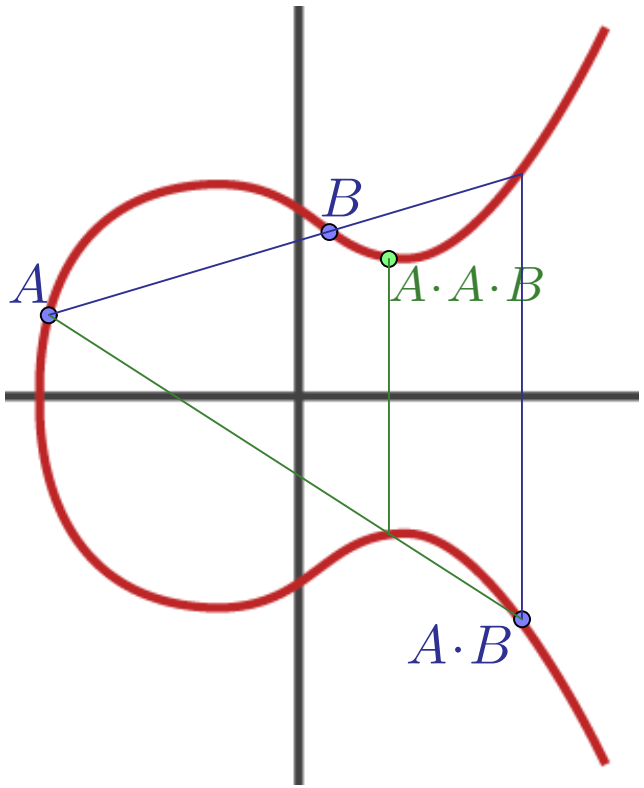

- przecina oś $y$ w pktach $\sqrt{b}$ i $-\sqrt{b}$;
- symetryczna względem osi $x$;
- dowolną prostą przecina w $\leqslant 3$ punktach;
- niech $A$ i $B$ leżą na krzywej w taki sposób, że ma ona jeszcze jeden punkt przecięcia z prostą $AB$; przez $A \cdot B$ rozumiemy punkt krzywej symetryczny do niego wzgl. osi $x$;
- operację $\cdot$ można stosować wielokrotnie:
  $A \cdot B$
  $A \cdot A \cdot B$
  $A \cdot A \cdot A \cdot B$
  ...

**Łatwe:** znając krzywą, jej punkty $A$ i $B$ oraz $n \in \mathbb{N}$, obliczyć $\underbrace{A \cdot \ldots \cdot A}_{n} \cdot B$

# Kryptografia krzywych eliptycznych

## Krzywa eliptyczna:

$$y^2 = x^3 + a \cdot x + b$$




- przecina oś $y$ w pktach $\sqrt{b}$ i $-\sqrt{b}$;
- symetryczna względem osi $x$;
- dowolną prostą przecina w $\leqslant 3$ punktach;
- niech $A$ i $B$ leżą na krzywej w taki sposób, że ma ona jeszcze jeden punkt przecięcia z prostą $AB$; przez $A \cdot B$ rozumiemy punkt krzywej symetryczny do niego wzgl. osi $x$;
- operację $\cdot$ można stosować wielokrotnie:
  $A \cdot B$
  $A \cdot A \cdot B$
  $A \cdot A \cdot A \cdot B$
  ...

**Łatwe:** znając krzywą, jej punkty $A$ i $B$ oraz $n \in \mathbb{N}$, obliczyć $\underbrace{A \cdot \ldots \cdot A}_{n} \cdot B$

**Trudne:** znając krzywą, jej punkty $A$ i $B$ oraz $\underbrace{A \cdot \ldots \cdot A}_{n} \cdot B$, znaleźć $n \in \mathbb{N}$

# Kryptografia krzywych eliptycznych

## Łatwe?

# Kryptografia krzywych eliptycznych

## Łatwe?

**Znamy:**

krzywą $y^2 = x^3 + ax + b$

punkt $A = (x_A, y_A)$ na tej krzywej

punkt $B = (x_B, y_B)$ na tej krzywej

# Kryptografia krzywych eliptycznych

**Łatwe?**

**Znamy:**

krzywą $y^2 = x^3 + ax + b$

punkt $A = (x_A, y_A)$ na tej krzywej

punkt $B = (x_B, y_B)$ na tej krzywej

**Szukamy:**

trzeci punkt przecięcia prostej $AB$ z krzywą

# Kryptografia krzywych eliptycznych

**Łatwe?**

**Znamy:**
krzywą $y^2 = x^3 + ax + b$
punkt $A = (x_A, y_A)$ na tej krzywej
punkt $B = (x_B, y_B)$ na tej krzywej

**Szukamy:**
trzeci punkt przecięcia prostej $AB$ z krzywą

**Równanie prostej $AB$ niepionowej i niepoziomej:**

$$p\,x + q\,y = 1$$

# Kryptografia krzywych eliptycznych

**Łatwe?**

**Znamy:**
krzywą $y^2 = x^3 + ax + b$
punkt $A = (x_A, y_A)$ na tej krzywej
punkt $B = (x_B, y_B)$ na tej krzywej

**Szukamy:**
trzeci punkt przecięcia prostej $AB$ z krzywą

**Równanie prostej $AB$ niepionowej i niepoziomej:**

$$px + qy = 1 \quad \text{gdzie} \quad \begin{cases} p\,x_A + q\,y_A = 1 \\ p\,x_B + q\,y_B = 1 \end{cases}$$

# Kryptografia krzywych eliptycznych

**Łatwe?**

**Znamy:**
krzywą $y^2 = x^3 + ax + b$
punkt $A = (x_A, y_A)$ na tej krzywej
punkt $B = (x_B, y_B)$ na tej krzywej

**Szukamy:**
trzeci punkt przecięcia prostej $AB$ z krzywą

**Równanie prostej $AB$ niepionowej i niepoziomej:**

$$px + qy = 1 \quad \text{gdzie} \quad \begin{cases} p\,x_A + q\,y_A = 1 \\ p\,x_B + q\,y_B = 1 \end{cases}$$

**Punkty przecięcia prostej z krzywą:**

$$\begin{cases} p\,x + q\,y = 1 \\ y^2 = x^3 + a \cdot x + b \end{cases}$$

# Kryptografia krzywych eliptycznych

**Łatwe?**

**Znamy:**
krzywą $y^2 = x^3 + ax + b$
punkt $A = (x_A, y_A)$ na tej krzywej
punkt $B = (x_B, y_B)$ na tej krzywej

**Szukamy:**
trzeci punkt przecięcia prostej $AB$ z krzywą

**Równanie prostej $AB$ niepionowej i niepoziomej:**

$$px + qy = 1 \quad \text{gdzie} \quad \begin{cases} p\,x_A + q\,y_A = 1 \\ p\,x_B + q\,y_B = 1 \end{cases}$$

**Punkty przecięcia prostej z krzywą:**

$$\begin{cases} p\,x + q\,y = 1 \\ y^2 = x^3 + a \cdot x + b \end{cases}$$

**Odbicie symetryczne wzgl. osi $x$:**

— zmienić znak współrzędnej $y$

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 \\ 3 \cdot 4 + 2 \equiv_{13} 1 \end{cases}$ — więc $A$ na prostej

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 & \text{— więc } A \text{ na prostej} \\ 3 \cdot 4 + 2 \equiv_{13} 1 & \text{— więc } B \text{ na prostej} \end{cases}$

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 & \text{— więc } A \text{ na prostej} \\ 3 \cdot 4 + 2 \equiv_{13} 1 & \text{— więc } B \text{ na prostej} \end{cases}$

**Przecięcie z krzywą:**

$(5, 12)$

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 & \text{— więc } A \text{ na prostej} \\ 3 \cdot 4 + 2 \equiv_{13} 1 & \text{— więc } B \text{ na prostej} \end{cases}$

**Przecięcie z krzywą:**

$(5, 12)$ bo $\begin{cases} 3 \cdot 5 + 12 \end{cases}$

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 & \text{— więc } A \text{ na prostej} \\ 3 \cdot 4 + 2 \equiv_{13} 1 & \text{— więc } B \text{ na prostej} \end{cases}$

**Przecięcie z krzywą:**

$(5, 12)$ bo $\begin{cases} 3 \cdot 5 + 12 \equiv_{13} 1 \end{cases}$

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 & \text{— więc } A \text{ na prostej} \\ 3 \cdot 4 + 2 \equiv_{13} 1 & \text{— więc } B \text{ na prostej} \end{cases}$

**Przecięcie z krzywą:**

$(5, 12)$ bo $\begin{cases} 3 \cdot 5 + 12 \equiv_{13} 1 & \text{— więc na prostej} \end{cases}$

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 & \text{— więc } A \text{ na prostej} \\ 3 \cdot 4 + 2 \equiv_{13} 1 & \text{— więc } B \text{ na prostej} \end{cases}$

**Przecięcie z krzywą:**

$(5, 12)$ bo $\begin{cases} 3 \cdot 5 + 12 \equiv_{13} 1 & \text{— więc na prostej} \\ 5^3 + 5 + 1 \end{cases}$

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 & \text{— więc } A \text{ na prostej} \\ 3 \cdot 4 + 2 \equiv_{13} 1 & \text{— więc } B \text{ na prostej} \end{cases}$

**Przecięcie z krzywą:**

$(5, 12)$ bo $\begin{cases} 3 \cdot 5 + 12 \equiv_{13} 1 & \text{— więc na prostej} \\ 5^3 + 5 + 1 \equiv_{13} 25 \cdot 5 + 6 \end{cases}$

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 & \text{— więc } A \text{ na prostej} \\ 3 \cdot 4 + 2 \equiv_{13} 1 & \text{— więc } B \text{ na prostej} \end{cases}$

**Przecięcie z krzywą:**

$(5, 12)$ bo $\begin{cases} 3 \cdot 5 + 12 \equiv_{13} 1 & \text{— więc na prostej} \\ 5^3 + 5 + 1 \equiv_{13} 25 \cdot 5 + 6 \\ \qquad\qquad \equiv_{13} (-1) \cdot 5 + 6 \end{cases}$

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 & \text{— więc } A \text{ na prostej} \\ 3 \cdot 4 + 2 \equiv_{13} 1 & \text{— więc } B \text{ na prostej} \end{cases}$

**Przecięcie z krzywą:**

$(5, 12)$ bo $\begin{cases} 3 \cdot 5 + 12 \equiv_{13} 1 & \text{— więc na prostej} \\ 5^3 + 5 + 1 \equiv_{13} 25 \cdot 5 + 6 & \\ \qquad\qquad \equiv_{13} (-1) \cdot 5 + 6 & \\ \qquad\qquad \equiv_{13} 1 & \end{cases}$

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 & \text{— więc } A \text{ na prostej} \\ 3 \cdot 4 + 2 \equiv_{13} 1 & \text{— więc } B \text{ na prostej} \end{cases}$

**Przecięcie z krzywą:**

$(5, 12)$ bo $\begin{cases} 3 \cdot 5 + 12 \equiv_{13} 1 & \text{— więc na prostej} \\ 5^3 + 5 + 1 \equiv_{13} 25 \cdot 5 + 6 & \\ \equiv_{13} (-1) \cdot 5 + 6 & \\ \equiv_{13} 1 \equiv_{13} 12^2 & \end{cases}$

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 & \text{— więc } A \text{ na prostej} \\ 3 \cdot 4 + 2 \equiv_{13} 1 & \text{— więc } B \text{ na prostej} \end{cases}$

**Przecięcie z krzywą:**

$(5, 12)$ bo $\begin{cases} 3 \cdot 5 + 12 \equiv_{13} 1 & \text{— więc na prostej} \\ 5^3 + 5 + 1 \equiv_{13} 25 \cdot 5 + 6 & \\ \qquad\qquad \equiv_{13} (-1) \cdot 5 + 6 & \\ \qquad\qquad \equiv_{13} 1 \equiv_{13} 12^2 & \text{— więc na krzywej} \end{cases}$

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 & \text{— więc } A \text{ na prostej} \\ 3 \cdot 4 + 2 \equiv_{13} 1 & \text{— więc } B \text{ na prostej} \end{cases}$

**Przecięcie z krzywą:**

$(5, 12)$ bo $\begin{cases} 3 \cdot 5 + 12 \equiv_{13} 1 & \text{— więc na prostej} \\ 5^3 + 5 + 1 \equiv_{13} 25 \cdot 5 + 6 & \\ \qquad\qquad \equiv_{13} (-1) \cdot 5 + 6 & \\ \qquad\qquad \equiv_{13} 1 \equiv_{13} 12^2 & \text{— więc na krzywej} \end{cases}$

**Odbicie symetryczne:**

$A \cdot B = (5, -12)$

# Kryptografia krzywych eliptycznych

Tak samo, ale zamiast na płaszczyźnie rzeczywistej $\mathbb{R} \times \mathbb{R}$, liczymy w $\mathbb{Z}_n \times \mathbb{Z}_n$ dla jakiejś liczby pierwszej $n$.

**Przykład:**

**Dane:**

$n = 13$ krzywa $y^2 \equiv_{13} x^3 + x + 1$

$A = (0, 1)$ — na krzywej, bo $0^3 + 0 + 1 \equiv_{13} 1 \equiv_{13} 1^2$

$B = (4, 2)$ — na krzywej, bo $4^3 + 4 + 1 \equiv_{13} 64 + 5 \equiv_{13} 17 \equiv_{13} 4 \equiv_{13} 2^2$

**Prosta $AB$:**

$3x + y \equiv_{13} 1$ bo $\begin{cases} 3 \cdot 0 + 1 \equiv_{13} 1 & \text{— więc } A \text{ na prostej} \\ 3 \cdot 4 + 2 \equiv_{13} 1 & \text{— więc } B \text{ na prostej} \end{cases}$

**Przecięcie z krzywą:**

$(5, 12)$ bo $\begin{cases} 3 \cdot 5 + 12 \equiv_{13} 1 & \text{— więc na prostej} \\ 5^3 + 5 + 1 \equiv_{13} 25 \cdot 5 + 6 & \\ \qquad\qquad \equiv_{13} (-1) \cdot 5 + 6 & \\ \qquad\qquad \equiv_{13} 1 \equiv_{13} 12^2 & \text{— więc na krzywej} \end{cases}$

**Odbicie symetryczne:**

$A \cdot B = (5, -12) \equiv_{13} (5, 1)$