Projekt 2

Autorzy: Sebastian Pergała, Michał Matuszyk

Zadanko 1.

(1) Niech $\mathbb{V}$ będzie dowolnym niepustym podzbiorem przestrzeni liniowej $\mathbb{K}^n$ nad ciałem $\mathbb{K}$.
Niech $x = (x_1, x_{2,} \dots, x_n), y = (y_1, y_{2,} \dots, y_n) \in \mathbb{V}$. Funkcję $D\colon \mathbb{V} \times \mathbb{V} \to \mathbb{R}$, określoną wzorem:

$$D(x,y) = |\{i \in [n]\colon x_i \neq y_i\}|$$

nazywamy **odległością Hamminga**.

(2) Funkcję $D\colon \mathbb{V} \times \mathbb{V} \to \mathbb{R}$, nazywamy **metryką** na zbiorze $\mathbb{V}$, jeśli spełnia następujące warunki:

- $D(x,y) = 0 \iff x = y$
- $D(x,y) = D(y,x)$
- $D(x,z) \leq D(x,y) + D(y,z)$.

Teza: Odległość Hamminga jest metryką.

Dowód:

Aby udowodnić, że odległość Hamminga jest metryką należy sprawdzić, czy spełnia warunki z (2).

Niech funkcja D, określona tak jak w (1), będzie odległością Haminga.

1) $\boldsymbol{D(x,y) = 0 \iff x = y}$

Udowodnię wynikanie $' \Longrightarrow ' \; i \; ' \Longleftarrow '$.

$(\Longrightarrow)\; D(x,y) = 0 \implies x = y$

Przekształcam równoważnie:

$$D(x,y) = 0 \implies x = y$$

$$|\{i \in [n]\colon x_i \neq y_i\}| = 0 \implies (x_1, x_{2,} \dots, x_n) = (y_1, y_{2,} \dots, y_n)$$

$$|\{i \in [n]\colon x_i \neq y_i\}| = 0 \implies \forall\, i \in [n]\; x_i = y_i$$

$$|\{i \in [n]\colon x_i \neq y_i\}| = 0 \implies {\sim}\exists\, i \in [n]\; x_i \neq y_i$$

Takich $i \in [n]$, że $x_i \neq y_i$ jest 0, więc $\sim\exists\, i \in [n]\; x_i \neq y_i$.

$(\Longleftarrow)\; x = y \implies D(x,y) = 0$

Przekształcam równoważnie:

$$x = y \Longrightarrow D(x,y) = 0$$

$$(x_1, x_2, \dots, x_n) = (y_1, y_2, \dots, y_n) \Longrightarrow |\{i \in [n]: x_i \neq y_i\}| = 0$$

$$\forall\, i \in [n]\ x_i = y_i \Longrightarrow |\{i \in [n]: x_i \neq y_i\}| = 0$$

$$\sim\exists\, i \in [n]\ x_i \neq y_i \Longrightarrow |\{i \in [n]: x_i \neq y_i\}| = 0$$

Skoro $\sim\exists$ i $\in$ [n] $x_i \neq y_i$, to takich i $\in$ [n], że $x_i \neq y_i$ nie ma, więc $|\{i \in [n]: x_i \neq y_i\}| = 0$.

Zatem zachodzi $' \Longrightarrow '$ *i* $' \Longleftarrow '$, *więc obie strony są sobie równoważne.*

2) $\boldsymbol{D(x,y) = D(y,x)}$

Przekształcam równoważnie:

$$D(x,y) = D(y,x)$$

$$|\{i \in [n]: x_i \neq y_i\}| = |\{i \in [n]: y_i \neq x_i\}|$$

Relacja nierówności '≠' jest symetryczna, czyli:

$$a \neq b \Longrightarrow b \neq a.$$

Z tego wynika, że $x_i \neq y_i \Longrightarrow y_i \neq x_i$

i analogicznie $y_i \neq x_i \Longrightarrow x_i \neq y_i$,

co się sprowadza do $x_i \neq y_i \Longleftrightarrow y_i \neq x_i$,

a więc $|\{i \in [n]: x_i \neq y_i\}| = |\{i \in [n]: y_i \neq x_i\}|$.

3) $\boldsymbol{D(x,z) \leq D(x,y) + D(y,z)}$

Ustalmy $A_{xz} := \{i \in [n]: x_i \neq z_i\}$, $A_{xy} := \{i \in [n]: x_i \neq y_i\}$, $A_{yz} := \{i \in [n]: y_i \neq z_i\}$

oraz $B_{xz} := \{i \in [n]: x_i = z_i\}$, $B_{xy} := \{i \in [n]: x_i = y_i\}$, $B_{yz} := \{i \in [n]: y_i = z_i\}$.

Wtedy $|A_{xz}| = n - |B_{xz}|$, $|A_{xy}| = n - |B_{xy}|$, $|A_{yz}| = n - |B_{yz}|$.

Przekształcam równoważnie:

$$D(x,z) \leq D(x,y) + D(y,z)$$

$$|\{i \in [n]: x_i \neq z_i\}| \leq |\{i \in [n]: x_i \neq y_i\}| + |\{i \in [n]: y_i \neq z_i\}|$$

$$|A_{xz}| \leq |A_{xy}| + |A_{yz}|$$

$$n - |B_{xz}| \leq n - |B_{xy}| + n - |B_{yz}|$$

$$|B_{xy}| + |B_{yz}| \leq n + |B_{xz}|.$$

Oznaczmy przez L lewą stronę nierówności $|B_{xy}| + |B_{yz}| \leq n + |B_{xz}|$, a przez P prawą stronę tej nierówności. Niech $L = \sum_{i=1}^{n} L_i$ i $P = \sum_{i=1}^{n} P_i$, gdzie L_i to liczba zbiorów po lewej stronie nierówności, do których należy dane i, a P_i to liczba zbiorów po prawej stronie nierówności, do których należy dane i powiększona o 1 (1 otrzymujemy przez równe podzielenie n po prawej stronie nierówności na liczbę rozważanych przypadków, których jest n).

Aby mieć pewność, że dana nierówność będzie spełniona wystarczy wykazać, że $\forall\, i \in [n]\ L_i \leq P_i + 1$. Postawiony warunek jest równoważny warunkowi: $\sim\exists\, i \in [n]\ L_i > P_i + 1$.

Ustalmy pewne $i \in [n]$.

$$L_i > P_i + 1 \Leftrightarrow i \in B_{xz} \wedge i \in B_{xy} \wedge i \notin B_{yz}$$

$$L_i > P_i + 1 \Leftrightarrow x_i = z_i \wedge x_i = y_i \wedge y_i \neq z_i$$

Relacja równoważności jest przechodnia i zwrotna, więc mamy:

$$L_i > P_i + 1 \Leftrightarrow x_i = z_i = y_i \wedge z_i \neq y_i$$

Zdanie $x_i = z_i = y_i \wedge z_i \neq y_i$ jest fałszywe dla ustalonego i, a więc równoważność $L_i > P_i + 1 \Leftrightarrow x_i = z_i = y_i \wedge z_i \neq y_i$ nie zachodzi.

To oznacza, że $\sim\exists\, i \in [n]\ L_i > P_i + 1 \Leftrightarrow x_i = z_i = y_i \wedge z_i \neq y_i$,

czyli $\sim\exists\, i \in [n]\ L_i > P_i + 1$. ■

Zadanko 2

Niech:

- $\mathbb{K}$ oznacza alfabet
- $B = \{e_1, e_2, \cdots, e_k\}$ *jest bazą kodu* C
- $G = \begin{pmatrix} e_1^T \\ e_2^T \\ \vdots \\ e_k^T \end{pmatrix}$ *oznacza macierz generująca kod* C
- $v = (v_1, v_2, \dots, v_k)^T = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_k \end{pmatrix}$ dowolny wektor należący do $\mathbb{K}^k$
- $e_i = \begin{pmatrix} e_{i,1} \\ e_{i,2} \\ \vdots \\ e_{i,k} \end{pmatrix}$ i-ty wektor z bazy kodu C
- $w = wektor\ powstały\ z\ kodowania\ wektora\ v$

Czyli $G = \begin{pmatrix} e_1^T \\ e_2^T \\ \vdots \\ e_k^T \end{pmatrix} = \begin{pmatrix} e_{1,1} & \cdots & e_{1,k} \\ \vdots & \ddots & \vdots \\ e_{k,1} & \cdots & e_{k,k} \end{pmatrix}$

Teza: Dla dowolnego (n, k) – kodu liniowego $\mathcal{C}$ nad skończonym ciałem $\mathbb{K}$ i jego macierzy generującej G powstałej z bazy kodu B wynikiem kodowania dowolnego wektora $v \in \mathbb{K}^k$ jest słowo kodowe kodu C.

Dowód:

Należy wykazać, że wektor w, jest kombinacją liniową wektorów bazy. Jeśli to wykażemy, to będzie oznaczać, że wektor w jest słowem kodowym kodu C.

Aby udowodnić ten fakt zakodujemy wektor v:

Kodowaniem wektora $v \in \mathbb{K}^v$ wprost z definicji, nazywamy wektor $w \in \mathbb{K}^v, taki, że$:

$$w = (v^T \cdot G)^T$$

$$w = (v^T \cdot G)^T = \left((v_1, v_2, \dots, v_k) \cdot \begin{pmatrix} e_{1,1} & \cdots & e_{1,k} \\ \vdots & \ddots & \vdots \\ e_{k,1} & \cdots & e_{k,k} \end{pmatrix} \right)^T = \begin{pmatrix} v_1 e_{1,1} + v_2 e_{2,1} + \cdots + v_k e_{k,1} \\ v_1 e_{1,2} + v_2 e_{2,2} + \cdots + v_k e_{k,2} \\ \vdots \\ v_1 e_{1,k} + v_2 e_{2,k} + \cdots + v_k e_{k,k} \end{pmatrix}$$

Udowodniliśmy, że wektor w jest kombinacją liniową wektorów z bazy, zatem wektor w jest wektorem z C, czyli słowem kodowym kodu C.

■

Zadanko 3.

Teza: Dla dowolnego (n, k) – kodu liniowego $\mathcal{C}$ nad skończonym ciałem $\mathbb{K}$ i jego macierzy generującej G powstałej z bazy kodu B algorytm *MinimizeHammingDistance* użyty do dekodowania słowa kodowego $w \in \mathcal{C}$ zwróci taki wektor $v' \in \mathbb{K}^k$, który w wyniku zakodowania go z użyciem macierzy G da wektor w.

Dowód:

Algorytm *MinimizeHammingDistance* wygląda następująco:

MinimizeHammingDistance($\mathcal{C}$, B, w){

#dane: $\mathcal{C}$ to (n, k) – kod liniowy nad ciałem $\mathbb{K}$, B to baza kodu $\mathcal{C}$, v to dekodowany wektor

#funkcja D jest określona następująco: dla $x = (x_1, x_2, \dots, x_n), y = (y_1, y_2, \dots, y_n)$

$\#D(x, y) = |\{i \in [n]: x_i \neq y_i\}|$

$m = min\{D(w, v_{\mathcal{C}}): v_{\mathcal{C}} \in \mathcal{C}\}$

$L = \{v_{\mathcal{C}} \in \mathcal{C}: D(w, v_{\mathcal{C}}) = m\}$

$v_{\mathcal{C}}$ = losowo wybrany wektor należący do L

v' = wektor współczynników wektora $v_{\mathcal{C}}$ w bazie B

wynik: wektor $v' \in \mathbb{K}^k$ }

Niech $\mathcal{C}$ będzie podprzestrzenią liniową wymiaru k przestrzeni linowej $\mathbb{K}^n$.

Niech $B \coloneqq \{e_1, e_2, e_3, \cdots, e_k\}$ będzie bazą Kodu $\mathcal{C}$ wymiaru k.

Niech $G \coloneqq \begin{pmatrix} e_1^T \\ e_2^T \\ \vdots \\ e_k^T \end{pmatrix}$. Niech $v = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_k \end{pmatrix}$, $v \in \mathbb{K}^k$. Niech $w \coloneqq (v^T \cdot G)^T$, $w \in \mathcal{C}$, $w \in \mathbb{K}^n$.

Chcemy udowodnić, że wynikowy wektor v' podanego wyżej algorytmu *MinimizeHammingDistance* po zakodowaniu za pomocą macierzy G da wektor w.

Postępując według algorytmu:

$\boldsymbol{m = min\{D(w, v_{\mathcal{C}}): v_{\mathcal{C}} \in \mathcal{C}\} = ?}$

$D(x, y)$ jest metryką (co zostało udowodnione w zadanku 1.) więc:

$$D(x, y) \geq 0 \quad \text{oraz} \quad D(x, y) = 0 \iff x = y.$$

Z racji, że $w \in \mathcal{C}$ i $v_{\mathcal{C}} \in \mathcal{C}$:

$$m = min\{D(w, v_{\mathcal{C}}): v_{\mathcal{C}} \in \mathcal{C}\} = D(w, w) = 0.$$

$\boldsymbol{L = \{v_c \in C:\ D(w, v_c) = m\}}$ **= ?**

m = 0, więc $L = \{v_c \in C:\ D(w, v_c) = 0\}$.

$D(x, y)$ jest metryką, więc $D(x, y) = 0 \iff x = y$.

Zatem $L = \{w\}$.

v_c = losowo wybrany wektor należący do L

|L| = 1, więc jest tylko jedna możliwość wyboru elementu ze zbioru L. Tą możliwością jest wybór wektora w.

Z tego wynika, że $v_c = w$.

v' = wektor współczynników wektora v_c w bazie B

W poprzednim kroku ustaliliśmy, że $v_c = w$, a więc

v' = wektor współczynników wektora w w bazie B

Wektor w jest wynikiem kodowania wektora v za pomocą macierzy generującej G.

$$w = (v^T \cdot G)^T = G^T \cdot v = B \cdot v = a_1 \cdot e_1 + a_2 \cdot e_2 + \cdots + a_k \cdot e_k = B \cdot v'$$

Z tego wynika, że $v' = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_k \end{pmatrix}$. Zatem $v' = v$.

Skoro $w = (v^T \cdot G)^T$ i $v' = v$, to $w = ((v')^T \cdot G)^T$.

Więc wynikowy wektor v' algorytmu *MinimizeHammingDistance* po zakodowaniu za pomocą macierzy G da wektor w.■

Zadanko 4

Niech

- $v, u, x \in \mathbb{K}^n$;
  - $v = (v_1, v_2, \dots, v_n)$
  - $u = (u_1, u_2, \dots, u_n)$
  - $x = (x_1, x_2, \dots, x_n)$, gdzie $v_i, u_1, x_1 \in \mathbb{K}$
- $d(u, v) = |\{i \in [n] : u_i \neq v_i\}| \rightarrow odległość\ Hamminga$
- $y \in \mathbb{K}$

Teza: Dla dowolnej przestrzeni liniowej $\mathbb{V}$ nad ciałem $\mathbb{K}$ odległość Hamminga jest niezmiennicza ze względu na przesunięcia.

Dowód:

Należy wykazać, że dla dowolnych wektorów $u, v, x \in \mathbb{K}^n$ odległość Hamminga słów u i v jest taka sama jak odległość słów $u + x\ i\ v + x$.

Prosto z definicji *odległości Hamminga* mamy:

$$d(u, v) = |\{i \in [n] : u_i \neq v_i\}| \triangleq |\{i \in [n] : u_i + y \neq v_i + y\}| =$$

$$= |\{i \in [n] : u_i + x_i \neq v_i + x_i\}| = d(u + x, v + x)$$

Δ - Jak wiemy, jeśli a, b, y ∈ $\mathbb{K}$ oraz $a = b,\ to\ a + y = b + y$
z czego wynika: jeśli a, b, y ∈ $\mathbb{K}$ oraz $a \neq b,\ to\ a + y \neq b + y.$

$Zatem\ wykazaliśmy, że\ dla\ odległość\ Hamminga\ jest\ niezmiennicza\ za\ względu\ na\ przesunięcia.$

■

Zadanko 5.

Obliczanie odległości Hamminga dla wektorów $\begin{pmatrix}1\\2\\0\\1\end{pmatrix}$ ***i*** $\begin{pmatrix}0\\0\\0\\1\end{pmatrix}$:

Odległość Hamminga D dla wektorów $x = (x_1, x_2, \dots, x_n)$, $y = (y_1, y_2, \dots, y_n)$ jest zdefiniowana następująco:

$$D(x,y) = |\{i \in [n]: x_i \neq y_i\}|$$

Zatem D(x, y) jest równe liczbie zbiorów dwuelementowych o różnych elementach ze zbioru $\{\{x_1, y_1\}, \{x_2, y_2\}, \{x_3, y_3\}, \cdots, \{x_n, y_n\}\}$.

$$D\left(\begin{pmatrix}1\\2\\0\\1\end{pmatrix}, \begin{pmatrix}0\\0\\0\\1\end{pmatrix}\right) = ?$$

Liczę liczbę zbiorów dwuelementowych o różnych elementach ze zbioru {{1, 0}, {2, 0}, {0, 0}, {1, 1}}.

Zbiory {1, 0} i {2, 0} mają dwa różne elementy. Liczba takich zbiorów jest równa 2. Zatem

$$D\left(\begin{pmatrix}1\\2\\0\\1\end{pmatrix}, \begin{pmatrix}0\\0\\0\\1\end{pmatrix}\right) = 2.$$

Które z wektorów ze zbioru $\left\{\begin{pmatrix}1\\2\\1\\2\\0\end{pmatrix},\begin{pmatrix}1\\1\\1\\1\\1\end{pmatrix},\begin{pmatrix}0\\0\\2\\1\\1\end{pmatrix},\begin{pmatrix}2\\2\\2\\1\\0\end{pmatrix}\right\}$ są najbliżej siebie w sensie Hamminga?

Ponumeruję wektory ze zbioru następująco: $1:\begin{pmatrix}1\\2\\1\\2\\0\end{pmatrix}$, $2:\begin{pmatrix}1\\1\\1\\1\\1\end{pmatrix}$, $3:\begin{pmatrix}0\\0\\2\\1\\1\end{pmatrix}$, $4:\begin{pmatrix}2\\2\\2\\1\\0\end{pmatrix}$.

Odległość między wektorami będę liczył korzystając z funkcji *HammingDistance[]* w *Wolfram Mathematica*. Przedstawię odległość poszczególnych wektorów od siebie za pomocą macierzy, gdzie i-ty wiersz oznacza i-ty wektor, j-ta kolumna oznacza j-ty wektor i pole a_{ij} oznacza odległość między i-tym, a j-tym wektorem w sensie Hamminga. $(\mathrm{i},\mathrm{j}\in[4])$ Postępując zgodnie z opisem otrzymana macierz ma postać:

$$\begin{pmatrix}0&3&5&3\\3&0&3&4\\5&3&0&3\\3&4&3&0\end{pmatrix}$$

Biorąc pod uwagę tylko różne wektory (bo wektor jest odległy sam od siebie o 0) minimalna odległość między rozpatrywanymi wektorami z danego zbioru jest równa min{3, 4, 5} = 3. Tak odległe są od siebie wektory:

1. i 2.,

1. i 4.,

2. i 3.,

3. i 4.

Relacja odległości między dwoma wektorami jest relacją symetryczną (co widać po symetryczności otrzymanej macierzy), więc jeśli i-ty wektor jest odległy o 3 od j-tego wektora, to j-ty wektor jest też odległy o 3 od i-tego wektora.

Zadanko 6

Wygeneruj wszystkie słowa kodowe dla (5, 3) – kodu liniowego C nad ciałem $\mathbb{Z}_7$ takiego, że bazą kodu liniowego C jest:

$$B = \left(\begin{pmatrix} 1 \\ 0 \\ 0 \\ 2 \\ 4 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \\ 5 \\ 6 \end{pmatrix} \right).$$

Zbiór wszystkich wektorów:

[[0, 0, 0, 0, 0], [0, 0, 1, 5, 6], [0, 0, 2, 3, 5], [0, 0, 3, 1, 4], [0, 0, 4, 6, 3], [0, 0, 5, 4, 2], [0, 0, 6, 2, 1],
[0, 1, 0, 1, 0], [0, 1, 1, 6, 6], [0, 1, 2, 4, 5], [0, 1, 3, 2, 4], [0, 1, 4, 0, 3], [0, 1, 5, 5, 2], [0, 1, 6, 3, 1],
[0, 2, 0, 2, 0], [0, 2, 1, 0, 6], [0, 2, 2, 5, 5], [0, 2, 3, 3, 4], [0, 2, 4, 1, 3], [0, 2, 5, 6, 2], [0, 2, 6, 4, 1],
[0, 3, 0, 3, 0], [0, 3, 1, 1, 6], [0, 3, 2, 6, 5], [0, 3, 3, 4, 4], [0, 3, 4, 2, 3], [0, 3, 5, 0, 2], [0, 3, 6, 5, 1],
[0, 4, 0, 4, 0], [0, 4, 1, 2, 6], [0, 4, 2, 0, 5], [0, 4, 3, 5, 4], [0, 4, 4, 3, 3], [0, 4, 5, 1, 2], [0, 4, 6, 6, 1],
[0, 5, 0, 5, 0], [0, 5, 1, 3, 6], [0, 5, 2, 1, 5], [0, 5, 3, 6, 4], [0, 5, 4, 4, 3], [0, 5, 5, 2, 2], [0, 5, 6, 0, 1],
[0, 6, 0, 6, 0], [0, 6, 1, 4, 6], [0, 6, 2, 2, 5], [0, 6, 3, 0, 4], [0, 6, 4, 5, 3], [0, 6, 5, 3, 2], [0, 6, 6, 1, 1],
[1, 0, 0, 2, 4], [1, 0, 1, 0, 3], [1, 0, 2, 5, 2], [1, 0, 3, 3, 1], [1, 0, 4, 1, 0], [1, 0, 5, 6, 6], [1, 0, 6, 4, 5],
[1, 1, 0, 3, 4], [1, 1, 1, 1, 3], [1, 1, 2, 6, 2], [1, 1, 3, 4, 1], [1, 1, 4, 2, 0], [1, 1, 5, 0, 6], [1, 1, 6, 5, 5],
[1, 2, 0, 4, 4], [1, 2, 1, 2, 3], [1, 2, 2, 0, 2], [1, 2, 3, 5, 1], [1, 2, 4, 3, 0], [1, 2, 5, 1, 6], [1, 2, 6, 6, 5],
[1, 3, 0, 5, 4], [1, 3, 1, 3, 3], [1, 3, 2, 1, 2], [1, 3, 3, 6, 1], [1, 3, 4, 4, 0], [1, 3, 5, 2, 6], [1, 3, 6, 0, 5],
[1, 4, 0, 6, 4], [1, 4, 1, 4, 3], [1, 4, 2, 2, 2], [1, 4, 3, 0, 1], [1, 4, 4, 5, 0], [1, 4, 5, 3, 6], [1, 4, 6, 1, 5],
[1, 5, 0, 0, 4], [1, 5, 1, 5, 3], [1, 5, 2, 3, 2], [1, 5, 3, 1, 1], [1, 5, 4, 6, 0], [1, 5, 5, 4, 6], [1, 5, 6, 2, 5],
[1, 6, 0, 1, 4], [1, 6, 1, 6, 3], [1, 6, 2, 4, 2], [1, 6, 3, 2, 1], [1, 6, 4, 0, 0], [1, 6, 5, 5, 6], [1, 6, 6, 3, 5],
[2, 0, 0, 4, 1], [2, 0, 1, 2, 0], [2, 0, 2, 0, 6], [2, 0, 3, 5, 5], [2, 0, 4, 3, 4], [2, 0, 5, 1, 3], [2, 0, 6, 6, 2],
[2, 1, 0, 5, 1], [2, 1, 1, 3, 0], [2, 1, 2, 1, 6], [2, 1, 3, 6, 5], [2, 1, 4, 4, 4], [2, 1, 5, 2, 3], [2, 1, 6, 0, 2],
[2, 2, 0, 6, 1], [2, 2, 1, 4, 0], [2, 2, 2, 2, 6], [2, 2, 3, 0, 5], [2, 2, 4, 5, 4], [2, 2, 5, 3, 3], [2, 2, 6, 1, 2],
[2, 3, 0, 0, 1], [2, 3, 1, 5, 0], [2, 3, 2, 3, 6], [2, 3, 3, 1, 5], [2, 3, 4, 6, 4], [2, 3, 5, 4, 3], [2, 3, 6, 2, 2],
[2, 4, 0, 1, 1], [2, 4, 1, 6, 0], [2, 4, 2, 4, 6], [2, 4, 3, 2, 5], [2, 4, 4, 0, 4], [2, 4, 5, 5, 3], [2, 4, 6, 3, 2],
[2, 5, 0, 2, 1], [2, 5, 1, 0, 0], [2, 5, 2, 5, 6], [2, 5, 3, 3, 5], [2, 5, 4, 1, 4], [2, 5, 5, 6, 3], [2, 5, 6, 4, 2],
[2, 6, 0, 3, 1], [2, 6, 1, 1, 0], [2, 6, 2, 6, 6], [2, 6, 3, 4, 5], [2, 6, 4, 2, 4], [2, 6, 5, 0, 3], [2, 6, 6, 5, 2],
[3, 0, 0, 6, 5], [3, 0, 1, 4, 4], [3, 0, 2, 2, 3], [3, 0, 3, 0, 2], [3, 0, 4, 5, 1], [3, 0, 5, 3, 0], [3, 0, 6, 1, 6],
[3, 1, 0, 0, 5], [3, 1, 1, 5, 4], [3, 1, 2, 3, 3], [3, 1, 3, 1, 2], [3, 1, 4, 6, 1], [3, 1, 5, 4, 0], [3, 1, 6, 2, 6],
[3, 2, 0, 1, 5], [3, 2, 1, 6, 4], [3, 2, 2, 4, 3], [3, 2, 3, 2, 2], [3, 2, 4, 0, 1], [3, 2, 5, 5, 0], [3, 2, 6, 3, 6],
[3, 3, 0, 2, 5], [3, 3, 1, 0, 4], [3, 3, 2, 5, 3], [3, 3, 3, 3, 2], [3, 3, 4, 1, 1], [3, 3, 5, 6, 0], [3, 3, 6, 4, 6],
[3, 4, 0, 3, 5], [3, 4, 1, 1, 4], [3, 4, 2, 6, 3], [3, 4, 3, 4, 2], [3, 4, 4, 2, 1], [3, 4, 5, 0, 0], [3, 4, 6, 5, 6],
[3, 5, 0, 4, 5], [3, 5, 1, 2, 4], [3, 5, 2, 0, 3], [3, 5, 3, 5, 2], [3, 5, 4, 3, 1], [3, 5, 5, 1, 0], [3, 5, 6, 6, 6],
[3, 6, 0, 5, 5], [3, 6, 1, 3, 4], [3, 6, 2, 1, 3], [3, 6, 3, 6, 2], [3, 6, 4, 4, 1], [3, 6, 5, 2, 0], [3, 6, 6, 0, 6],
[4, 0, 0, 1, 2], [4, 0, 1, 6, 1], [4, 0, 2, 4, 0], [4, 0, 3, 2, 6], [4, 0, 4, 0, 5], [4, 0, 5, 5, 4], [4, 0, 6, 3, 3],
[4, 1, 0, 2, 2], [4, 1, 1, 0, 1], [4, 1, 2, 5, 0], [4, 1, 3, 3, 6], [4, 1, 4, 1, 5], [4, 1, 5, 6, 4], [4, 1, 6, 4, 3],
[4, 2, 0, 3, 2], [4, 2, 1, 1, 1], [4, 2, 2, 6, 0], [4, 2, 3, 4, 6], [4, 2, 4, 2, 5], [4, 2, 5, 0, 4], [4, 2, 6, 5, 3],
[4, 3, 0, 4, 2], [4, 3, 1, 2, 1], [4, 3, 2, 0, 0], [4, 3, 3, 5, 6], [4, 3, 4, 3, 5], [4, 3, 5, 1, 4], [4, 3, 6, 6, 3],
[4, 4, 0, 5, 2], [4, 4, 1, 3, 1], [4, 4, 2, 1, 0], [4, 4, 3, 6, 6], [4, 4, 4, 4, 5], [4, 4, 5, 2, 4], [4, 4, 6, 0, 3],
[4, 5, 0, 6, 2], [4, 5, 1, 4, 1], [4, 5, 2, 2, 0], [4, 5, 3, 0, 6], [4, 5, 4, 5, 5], [4, 5, 5, 3, 4], [4, 5, 6, 1, 3],
[4, 6, 0, 0, 2], [4, 6, 1, 5, 1], [4, 6, 2, 3, 0], [4, 6, 3, 1, 6], [4, 6, 4, 6, 5], [4, 6, 5, 4, 4], [4, 6, 6, 2, 3],
[5, 0, 0, 3, 6], [5, 0, 1, 1, 5], [5, 0, 2, 6, 4], [5, 0, 3, 4, 3], [5, 0, 4, 2, 2], [5, 0, 5, 0, 1], [5, 0, 6, 5, 0],
[5, 1, 0, 4, 6], [5, 1, 1, 2, 5], [5, 1, 2, 0, 4], [5, 1, 3, 5, 3], [5, 1, 4, 3, 2], [5, 1, 5, 1, 1], [5, 1, 6, 6, 0],

[5, 2, 0, 5, 6], [5, 2, 1, 3, 5], [5, 2, 2, 1, 4], [5, 2, 3, 6, 3], [5, 2, 4, 4, 2], [5, 2, 5, 2, 1], [5, 2, 6, 0, 0],
[5, 3, 0, 6, 6], [5, 3, 1, 4, 5], [5, 3, 2, 2, 4], [5, 3, 3, 0, 3], [5, 3, 4, 5, 2], [5, 3, 5, 3, 1], [5, 3, 6, 1, 0],
[5, 4, 0, 0, 6], [5, 4, 1, 5, 5], [5, 4, 2, 3, 4], [5, 4, 3, 1, 3], [5, 4, 4, 6, 2], [5, 4, 5, 4, 1], [5, 4, 6, 2, 0],
[5, 5, 0, 1, 6], [5, 5, 1, 6, 5], [5, 5, 2, 4, 4], [5, 5, 3, 2, 3], [5, 5, 4, 0, 2], [5, 5, 5, 5, 1], [5, 5, 6, 3, 0],
[5, 6, 0, 2, 6], [5, 6, 1, 0, 5], [5, 6, 2, 5, 4], [5, 6, 3, 3, 3], [5, 6, 4, 1, 2], [5, 6, 5, 6, 1], [5, 6, 6, 4, 0],
[6, 0, 0, 5, 3], [6, 0, 1, 3, 2], [6, 0, 2, 1, 1], [6, 0, 3, 6, 0], [6, 0, 4, 4, 6], [6, 0, 5, 2, 5], [6, 0, 6, 0, 4],
[6, 1, 0, 6, 3], [6, 1, 1, 4, 2], [6, 1, 2, 2, 1], [6, 1, 3, 0, 0], [6, 1, 4, 5, 6], [6, 1, 5, 3, 5], [6, 1, 6, 1, 4],
[6, 2, 0, 0, 3], [6, 2, 1, 5, 2], [6, 2, 2, 3, 1], [6, 2, 3, 1, 0], [6, 2, 4, 6, 6], [6, 2, 5, 4, 5], [6, 2, 6, 2, 4],
[6, 3, 0, 1, 3], [6, 3, 1, 6, 2], [6, 3, 2, 4, 1], [6, 3, 3, 2, 0], [6, 3, 4, 0, 6], [6, 3, 5, 5, 5], [6, 3, 6, 3, 4],
[6, 4, 0, 2, 3], [6, 4, 1, 0, 2], [6, 4, 2, 5, 1], [6, 4, 3, 3, 0], [6, 4, 4, 1, 6], [6, 4, 5, 6, 5], [6, 4, 6, 4, 4],
[6, 5, 0, 3, 3], [6, 5, 1, 1, 2], [6, 5, 2, 6, 1], [6, 5, 3, 4, 0], [6, 5, 4, 2, 6], [6, 5, 5, 0, 5], [6, 5, 6, 5, 4],
[6, 6, 0, 4, 3], [6, 6, 1, 2, 2], [6, 6, 2, 0, 1], [6, 6, 3, 5, 0], [6, 6, 4, 3, 6], [6, 6, 5, 1, 5], [6, 6, 6, 6, 4]]

Jest ich 343.

Kod użyty do ich generowania:

```
baza = [[1, 0, 0, 2, 4],
        [0, 1, 0, 1, 0],
        [0, 0, 1, 5, 6]]

kombinacje = []
mod = 7
for i in range(mod):  # tworzymy wszystkie możliwe kombinacje
    for j in range(mod):
        for d in range(mod):
            kombinacje.append([i, j, d])

slowa = []
for kombinacja in kombinacje:
    slowo = []
    for i in range(5):
        element = 0
        element += baza[0][i] * kombinacja[0]
        element += baza[1][i] * kombinacja[1]
        element += baza[2][i] * kombinacja[2]
        element %= mod
        slowo.append(element)
    if slowo not in slowa:  # sprawdzamy, czy dany wektor się nie powatrza
        slowa.append(slowo)

print(slowa)
```

Zadanko 7

Macierzą generującą kod (n, k) – kod liniowy $\mathcal{C}$ nazywamy macierz G o k wierszach i n kolumnach taką, że $G \coloneqq \begin{pmatrix} e_1^T \\ e_2^T \\ \vdots \\ e_k^T \end{pmatrix}$, gdzie $\{e_1, e_2, e_3, \cdots, e_k\}$ to wektory tworzące bazę kodu $\mathcal{C}$.
Dla kodu liniowego $\mathcal{C}$ może istnieć więcej niż jedna baza, więc może istnieć więcej niż jedna macierz generująca.

W zadanku 6 mamy (5, 3) – kod liniowy C nad ciałem $\mathbb{Z}_7$. Istnieje tyle macierzy generujących ten kod ile istnieje uporządkowanych baz tego kodu (bo kolejność wektorów ma znaczenie ze względu na definicję macierzy generującej).
Aby stworzyć macierz generującą należy najpierw wybrać niezerowy wektor v_1 należący do kodu C. Skoro wszystkich wektorów jest $7 \cdot 7 \cdot 7 = 343$, to niezerowych wektorów jest $7 \cdot 7 \cdot 7 - 1 = 342$.
Następnie po wybraniu wektora rozpinającego przestrzeń 1 – wymiarową wybieramy drugi niezerowy wektor, który nie jest pomnożonym przez skalar wektorem v_1. Skończone ciało $\mathbb{Z}_7$ ma 7 elementów (w tym 0), więc drugi wektor v_2 można wybrać na $343 - 7 = 336$ sposobów.
Pozostał do wybrania wektor v_3. Analogicznie, należy wybrać niezerowy wektor niebędący kombinacją liniową v_1 *i* v_2. Możemy to zrobić na $343 - 7 \cdot 7 = 294$ sposobów.
Wszystkich różnych macierzy generujących (5, 3) – kod liniowy C nad ciałem $\mathbb{Z}_7$ jest:
$(7^3 - 7^0) \cdot (7^3 - 7^0) \cdot (7^3 - 7^0) = 342 \cdot 336 \cdot 294 = 33\,784\,128$.

Dla przestrzeni $\mathcal{L}(e_1, e_2, , \cdots, e_k)$, gdzie $\{e_1, e_2, , \cdots, e_k\}$ to zbiór wektorów tworzących bazę kodu $\mathcal{C}$, można utworzyć macierz generującą G poprzez transponowanie macierzy $A \coloneqq (e_1, e_2, , \cdots, e_k)$. $G = A^T$. Wykonuję wspomnianą operację za pomocą następującego kodu w *Pythonie*:

```
# B = {e1, e2, e3}
# e1 = [1, 0, 0, 2, 4]
# e1 = [0, 1, 0, 1, 0]
# e1 = [0, 0, 1, 5, 6]
# Tworzę macierz A = [e1 e2 e3]
A = [[1, 0, 0, 2, 4],
     [0, 1, 0, 1, 0],
     [0, 0, 1, 5, 6]]
# G = A transpononowane
G = []
for i in range(len(A[0])):
    wiersz = []
    for j in range(len(A)):
        wiersz.append(A[j][i])
    G.append(wiersz)
```

Wynikiem jest macierz $G = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 2 & 1 & 5 \\ 4 & 0 & 6 \end{pmatrix}$.

Następnie będę dekodował dowolnie wybrany przez siebie wektor $v \in \mathbb{Z}_7^5$ za pomocą algorytmu *MinimizeHammingDistance.*

Niech $v_1 := \begin{pmatrix} 5 \\ 2 \\ 5 \\ 2 \\ 0 \end{pmatrix}, v_1 \notin \mathcal{C}$ oraz $v_2 := \begin{pmatrix} 5 \\ 2 \\ 5 \\ 2 \\ 1 \end{pmatrix}, v_2 \in \mathcal{C}$.

Za pomocą następującego kodu będę dekodował te wektory:

```
def odleglosc_Haminga(v, w):
    odleglosc = 0
    for i in range(len(v)):
        if (v[i] != w[i]):
            odleglosc += 1
    return odleglosc



def minimizeHammingDistance(C, B, v, mod):

    # m = min{d(v, w): wEC}
    m = float("inf")
    for slowo in C:
        odl = odleglosc_Haminga(slowo, v)
        if odl < m:
            m = odl

    # L = {wEC: d(v, w) = m}
    L = []
    for slowo in C:
        if odleglosc_Haminga(slowo, v) == m:
            L.append(slowo)

    # w - losowo wybrany wektor należący do L
    w = L[random.randint(0, len(L) - 1)]

    # r = wektor współczynników wektora 'w' w bazie B
    r = []
    for i in range(len(B)):
        wspolczynnik = w[i] / B[i][i]
        for j in range(len(w)):
            w[j] -= wspolczynnik * B[i][j]
            w[j] += mod * mod
            w[j] = w[j] % mod
        r.append(int(wspolczynnik))

    return r
```

Wynikiem powyższego kodu dla obu ustalonych wektorów jest wektor $\begin{pmatrix} 5 \\ 2 \\ 5 \end{pmatrix}$.

Zadanko 8

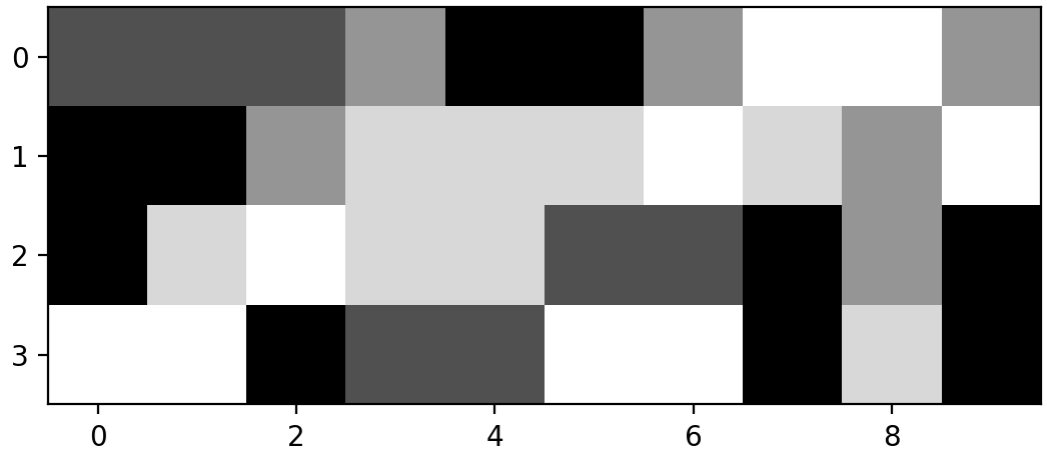
a) Wygeneruj losową macierz o 10 kolumnach i 4 wierszach o wyrazach z ciała $\mathbb{Z}_5$.
Wygenerowana macierz:

$$\begin{pmatrix} 3 & 3 & 3 & 2 & 4 & 4 & 2 & 0 & 0 & 2 \\ 4 & 4 & 2 & 1 & 1 & 1 & 0 & 1 & 2 & 0 \\ 4 & 1 & 0 & 1 & 1 & 3 & 3 & 4 & 2 & 4 \\ 0 & 0 & 4 & 3 & 3 & 0 & 0 & 4 & 1 & 4 \end{pmatrix}$$

b) Unormowana macierz:

$$\begin{pmatrix} 0{,}75 & 0{,}75 & 0{,}75 & 0{,}5 & 0{,}25 & 0{,}25 & 0{,}5 & 0 & 0 & 0{,}5 \\ 0{,}25 & 0{,}25 & 0{,}5 & 0{,}25 & 0{,}25 & 0{,}25 & 0 & 0{,}25 & 0{,}5 & 0 \\ 0{,}25 & 0{,}25 & 0 & 0{,}25 & 0{,}25 & 0{,}75 & 0{,}75 & 0{,}25 & 0{,}5 & 0{,}25 \\ 0 & 0 & 0{,}25 & 0{,}75 & 0{,}75 & 0 & 0 & 0{,}25 & 0{,}25 & 0{,}25 \end{pmatrix}$$

Obraz wygenerowany:



c)

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 4 & 4 & 2 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 3 & 0 & 2 & 2 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 2 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 4 & 3 & 0 \end{pmatrix}$$

Teza: Istnieje (11, 4) – kod liniowy nad ciałem ,taki że G jest jego macierzą generującą.
Dowód:
Niech:

$$e_1 = (1,0,0,0,0,4,4,2,0,1,1)$$
$$e_2 = (0,1,0,0,0,3,0,2,2,1,0)$$
$$e_3 = (0,0,1,0,0,2,0,1,1,1,1)$$
$$e_4 = (0,0,1,1,0,0,0,0,4,3,0)$$

Sprawdźmy, czy wektory e_1, e_2, e_3, e_4 są *liniowo niezależne*.

Zapisane w macierzy są zeschodkowane i widać, że są liniowo niezależne.
Zatem tworzą podprzestrzeń wymiaru 4 w ciele $\mathbb{Z}_5$.

■

d) Przykładowy wektor wybrany do zakodowania:

$$v_{doZakodowania} = (4,1,1,3)$$

Zakodowany wektor:

$$v_{zakodowany} = (4, 1, 1, 3, 3, 1, 1, 1, 0, 0, 0)$$

Wszystkie wektory zakodowane:

[[3, 4, 4, 0, 0, 2, 2, 3, 2, 1, 2],
[3, 4, 1, 0, 0, 1, 2, 0, 4, 3, 4],
[3, 2, 0, 4, 4, 3, 2, 0, 0, 2, 3],
[2, 1, 1, 3, 3, 3, 3, 2, 0, 3, 3],
[4, 1, 1, 3, 3, 1, 1, 1, 0, 0, 0],
[4, 1, 3, 0, 0, 0, 1, 3, 0, 3, 2],
[2, 0, 3, 0, 0, 4, 3, 2, 3, 0, 0],
[0, 1, 4, 4, 4, 1, 0, 1, 2, 2, 4],
[0, 2, 2, 1, 1, 0, 0, 1, 0, 2, 2],
[2, 0, 4, 4, 4, 1, 3, 3, 0, 3, 1]]

e) Wszystkie wektory po wysłaniu ich przez kanał:

[[3, 4, 4, 0, 0, 2, 2, 3, 2, 1, 2],
[3, 4, 1, 0, 0, 1, 2, 0, 4, 3, 4],
[3, 2, 0, 4, 4, 3, 2, 0, 0, 2, 3],
[2, 1, 1, 3, 3, 3, 3, 2, 0, 3, 3],
[4, 1, 1, 3, 3, 1, 1, 1, 3, 0, 0],
[2, 1, 3, 0, 0, 0, 1, 3, 0, 3, 2],
[2, 0, 3, 0, 0, 4, 3, 2, 3, 0, 0],
[0, 1, 4, 4, 4, 1, 0, 1, 2, 2, 4],
[0, 0, 2, 1, 1, 0, 0, 1, 3, 2, 2],
[2, 0, 4, 4, 4, 1, 3, 3, 0, 3, 1]]

f) Odkodowane wektory po przesłaniu przez kanał:

[[3, 4, 4, 0],
[3, 4, 1, 0],
[3, 2, 0, 4],
[2, 1, 1, 3],
[4, 1, 1, 3],
[4, 1, 3, 0],
[2, 0, 3, 0],
[0, 1, 4, 4],
[0, 2, 2, 1],
[2, 0, 4, 4]]

g) Macierz z wektorów z punktu f odpowiadająca macierzy z punktu a:

$$\begin{pmatrix} 3 & 3 & 3 & 2 & 4 & 4 & 2 & 0 & 0 & 2 \\ 4 & 4 & 2 & 1 & 1 & 1 & 0 & 1 & 2 & 0 \\ 4 & 1 & 0 & 1 & 1 & 3 & 3 & 4 & 2 & 4 \\ 0 & 0 & 4 & 3 & 3 & 0 & 0 & 4 & 1 & 4 \end{pmatrix}$$

h) Macierz z punktu a:

$$\begin{pmatrix} 3 & 3 & 3 & 2 & 4 & 4 & 2 & 0 & 0 & 2 \\ 4 & 4 & 2 & 1 & 1 & 1 & 0 & 1 & 2 & 0 \\ 4 & 1 & 0 & 1 & 1 & 3 & 3 & 4 & 2 & 4 \\ 0 & 0 & 4 & 3 & 3 & 0 & 0 & 4 & 1 & 4 \end{pmatrix}$$

Macierz z punktu g:

$$\begin{pmatrix} 3 & 3 & 3 & 2 & 4 & 4 & 2 & 0 & 0 & 2 \\ 4 & 4 & 2 & 1 & 1 & 1 & 0 & 1 & 2 & 0 \\ 4 & 1 & 0 & 1 & 1 & 3 & 3 & 4 & 2 & 4 \\ 0 & 0 & 4 & 3 & 3 & 0 & 0 & 4 & 1 & 4 \end{pmatrix}$$

Poprawnie odkodowanych zostało 10 kolumn (wszystkie kolumny).

i) Unormowana macierz:

$$\begin{pmatrix} 0{,}75 & 0{,}75 & 0{,}75 & 0{,}5 & 0{,}25 & 0{,}25 & 0{,}5 & 0 & 0 & 0{,}5 \\ 0{,}25 & 0{,}25 & 0{,}5 & 0{,}25 & 0{,}25 & 0{,}25 & 0 & 0{,}25 & 0{,}5 & 0 \\ 0{,}25 & 0{,}25 & 0 & 0{,}25 & 0{,}25 & 0{,}75 & 0{,}75 & 0{,}25 & 0{,}5 & 0{,}25 \\ 0 & 0 & 0{,}25 & 0{,}75 & 0{,}75 & 0 & 0 & 0{,}25 & 0{,}25 & 0{,}25 \end{pmatrix}$$

Obraz wygenerowany: