

Ćwiczenia 1

19.03.2025 r.

Rafał Leja 340879

1. Dla każdego z podanych poniżej adresów IP w notacji CIDR określ, czy jest to adres sieci, adres rozgłoszeniowy czy też adres komputera. W każdym przypadku wyznacz odpowiadający mu adres sieci, rozgłoszeniowy i jakiś adres IP innego komputera w tej samej sieci.

- 10.1.2.3/8
 - ☐ Adres sieci: 10.0.0.0
 - ☐ Adres rozgłoszeniowy: 10.255.255.255
 - ☒ **Adres komputera:** 10.1.2.3
- 156.17.0.0/16
 - ☒ **Adres sieci:** 156.17.0.0
 - ☐ Adres rozgłoszeniowy: 156.17.255.255
 - ☐ Adres komputera: 156.17.0.1
- 99.99.99.99/27
 - ☐ Adres sieci: 99.99.99.96 (**0110 0000**)
 - ☐ Adres rozgłoszeniowy: 99.99.99.127 (**0111 1111**)
 - ☒ **Adres komputera:** 99.99.99.99 (**0110 0011**)
- 156.17.64.4/30
 - ☒ **Adres sieci:** 156.17.64.4 (**0000 0100**)
 - ☐ Adres rozgłoszeniowy: 156.17.64.7 (**0000 0111**)
 - ☐ Adres komputera: 156.17.64.5 (**0000 0101**)
- 123.123.123.123/32
 - ☐ Adres sieci: -
 - ☐ Adres rozgłoszeniowy: -
 - ☒ **Adres komputera:** 123.123.123.123

2. Podziel sieć 10.10.0.0/16 na 5 rozłącznych podsieci, tak aby każdy z adresów IP z sieci 10.10.0.0/16 był w jednej z tych 5 podsieci. Jak zmieniła się liczba adresów IP możliwych do użycia przy adresowaniu komputerów? Jaki jest minimalny rozmiar podsieci, który możesz uzyskać w ten sposób?

1. 10.10.0.0/18: 2^{14} adresów IP
2. 10.10.64.0/18: 2^{14} adresów IP
3. 10.10.128.0/19: 2^{13} adresów IP
4. 10.10.160.0/19: 2^{13} adresów IP
5. 10.10.192.0/19: 2^{13} adresów IP

Liczba adresów IP możliwych do użycia przy adresowaniu komputerów zm-

niejszyła się z

$$2^{16} - 2 = 65534$$

do

$$2^{14} + 2^{14} + 2^{13} + 2^{13} + 2^{13} - 5 \cdot 2 = 57334$$

Minimalny rozmiar podsieci, który można uzyskać w ten sposób to $2^{13} = 8192$ adresy IP.

3. Tablica routingu zawiera następujące wpisy (podsieć → dokąd wysłać):

- 0.0.0.0/0 → do routera A
- 10.0.0.0/23 → do routera B
- 10.0.2.0/24 → do routera B
- 10.0.3.0/24 → do routera B
- 10.0.1.0/24 → do routera C
- 10.0.0.128/25 → do routera B
- 10.0.1.8/29 → do routera B
- 10.0.1.16/29 → do routera B
- 10.0.1.24/29 → do routera B

Napisz równoważną tablicę routingu zawierającą jak najmniej wpisów.

Rozpiszmy zakresy adresów IP dla podsieci B i C:

- B:

$$\begin{aligned} & [10.0.0.0, 10.0.1.255] \cup [10.0.2.0, 10.0.2.255] \cup [10.0.3.0, 10.0.3.255] \cup [10.0.0.128, 10.0.0.255] \cup \\ & [10.0.1.8, 10.0.1.15] \cup [10.0.1.16, 10.0.1.23] \cup [10.0.1.24, 10.0.1.31] = \\ & [10.0.0.0, 10.0.3.255] \Rightarrow 10.0.0.0/22 \end{aligned}$$

- C:

$$[10.0.1.0, 10.0.1.255] \setminus [10.0.1.8, 10.0.1.31]$$

Zoptymalizowana tablica:

- 0.0.0.0/0 → do routera A
- 10.0.3.0/22 → do routera B
- 10.0.1.0/24 → do routera C
- 10.0.1.8/29 → do routera B
- 10.0.1.16/28 → do routera B

4. Wykonaj powyższe zadanie dla tablicy:

- 0.0.0.0/0 → do routera A
- 10.0.0.0/8 → do routera B
- 10.3.0.0/24 → do routera C
- 10.3.0.32/27 → do routera B
- 10.3.0.64/27 → do routera B

- 10.3.0.96/27 → do routera B

Rozpiszmy zakresy adresów IP dla podsieci C:

$$\begin{aligned}
 & [10.3.0.0, 10.3.0.255] \setminus [10.3.0.32, 10.3.0.63] \setminus [10.3.0.64, 10.3.0.95] \setminus [10.3.0.96, 10.3.0.127] \\
 &= [10.3.0.0, 10.3.0.31] \cup [10.3.0.128, 10.3.0.255] \\
 &\Rightarrow 10.3.0.0/27; 10.3.0.128/25
 \end{aligned}$$

Zoptymalizowana tablica:

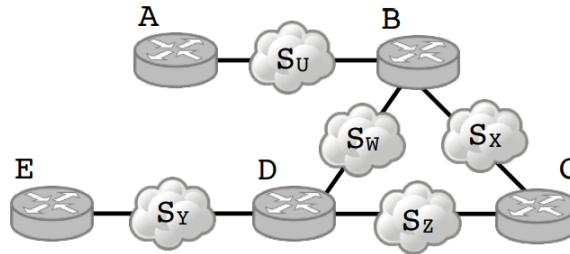
- 0.0.0.0/0 → do routera A
- 10.0.0.0/8 → do routera B
- 10.3.0.0/27 → do routera C
- 0.3.0.128/25 → do routera C

5. Jak uporządkować wpisy w tablicy routingu, żeby zasada najlepszego dopasowania odpowiadała wyborowi „pierwszy pasujący” (tj. przeglądaniu tablicy od początku do końca aż do momentu napotkania dowolnej pasującej reguły)? Odpowiedź uzasadnij formalnie.

Należy posortować wpisy w tablicy routingu w kolejności malejącej długości prefiksu.

Rozważmy adres IP oznaczony jako a . Niech x oraz y będą kolejnymi wpisami w tablicy routingu. Załóżmy że a pasuje do x na n bitach. Skoro wpisy są posortowane malejąco, to a nie pasuje do y na więcej niż n bitach. W przeciwnym przypadku y miałby dłuższy prefiks niż x .

6. W podanej niżej sieci tablice routingu budowane są za pomocą algorytmu wektora odległości. Pokaż (krok po kroku), jak będzie się to odbywać. W ilu krokach zostanie osiągnięty stan stabilny?



Krok 0:

sieci \ routery	A	B	C	D	E
SU	1	1	—	—	—
SW	—	1	—	1	—
SX	—	1	1	—	—
SY	—	—	—	1	1

sieci \ routery	A	B	C	D	E
SZ	—	—	1	1	—

Krok 1:

sieci \ routery	A	B	C	D	E
SU	1	1	2(B)	2(B)	—
SW	2(B)	1	2(B)	1	2(D)
SX	2(B)	1	1	2(C)	—
SY	—	2(D)	2(D)	1	1
SZ	—	2(C)	1	1	2(D)

Krok 2: -> **stan stabilny**

sieci \ routery	A	B	C	D	E
SU	1	1	2(B)	2(B)	3(D)
SW	2(B)	1	2(B)	1	2(D)
SX	2(B)	1	1	2(C)	3(D)
SY	3(B)	2(D)	2(D)	1	1
SZ	3(B)	2(C)	1	1	2(D)

7. Załóżmy, że w powyższej sieci tablice routingu zostały już zbudowane. Co będzie się działo (krok po kroku), jeśli zostanie dodana sieć SQ łącząca routery A i E?

Krok 0:

sieci \ routery	A	B	C	D	E
SU	1	1	2(B)	2(B)	3(D)
SW	2(B)	1	2(B)	1	2(D)
SX	2(B)	1	1	2(C)	3(D)
SY	3(B)	2(D)	2(D)	1	1
SZ	3(B)	2(C)	1	1	2(D)
SQ	1	—	—	—	1

Krok 1:

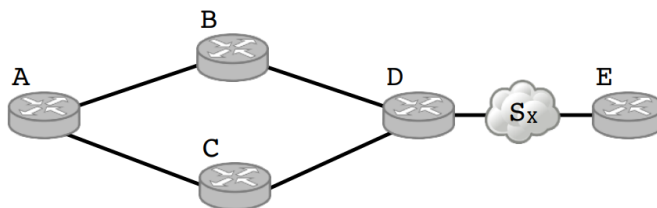
sieci \ routery	A	B	C	D	E
SU	1	1	2(B)	2(B)	3(D)
SW	2(B)	1	2(B)	1	2(D)

sieci \ routery	A	B	C	D	E
SX	2(B)	1	1	2(C)	3(D)
SY	3(B)	2(D)	2(D)	1	1
SZ	3(B)	2(C)	1	1	2(D)
SQ	1	2(A)	—	2(D)	1

Krok 2: -> **stan stabilny**

sieci \ routery	A	B	C	D	E
SU	1	1	2(B)	2(B)	3(D)
SW	2(B)	1	2(B)	1	2(D)
SX	2(B)	1	1	2(C)	3(D)
SY	3(B)	2(D)	2(D)	1	1
SZ	3(B)	2(C)	1	1	2(D)
SQ	1	2(A)	3(B)	2(D)	1

8. W przedstawionej poniżej sieci uszkodzeniu ulega połączenie między routerami D i E. Załóżmy, że w sieci działa algorytm wektora odległości wykorzystujący technikę zatruwania ścieżki zwrotnej (poison reverse). Pokaż — opisując krok po kroku jakie komunikaty są przesyłane między routerami — że może powstać cykl w routingu.



Zakładam że połączenie między D i Sx jest uszkodzone.

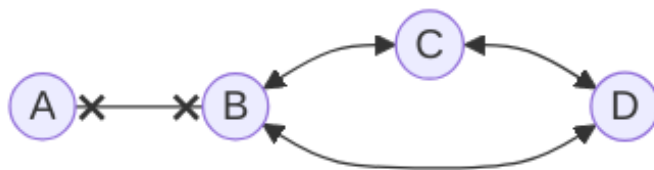
- **Poison reverse** - Jeśli router X jest wpisany jako następny router na ścieżce do S, to wysyłamy do X informację „mam do S ścieżkę nieskończoną“.

Komunikaty:

1. D -> B: „mam do SX ścieżkę nieskończoną“
2. B -> A: „mam do SX ścieżkę nieskończoną“
3. C -> A: „mam do SX 2 kroki, przez D”
4. A -> B: „mam do SX 3 kroki, przez C”
5. B -> D: „mam do SX 4 kroki, przez A”
6. D -> C: „mam do SX 5 kroków, przez B”
7. C -> A: „mam do SX 6 kroków, przez D”
8. ...

Krok	—	A	B	C	D	E
0	SX	3(B)	2(D)	2(D)	∞	∞
1	SX	3(B)	∞	2(D)	∞	∞
2	SX	∞	∞	2(D)	∞	∞
3	SX	3(C)	∞	2(D)	∞	∞
4	SX	3(C)	4(A)	2(D)	∞	∞
5	SX	3(C)	4(A)	2(D)	5(B)	∞
6	SX	3(C)	4(A)	6(D)	5(B)	∞
7	SX	7(C)	4(A)	6(D)	5(B)	∞
...						
N	SX	N(C)	N-3(A)	N-1(D)	N-2(B)	∞

9. Pokaż, że przy wykorzystaniu algorytmu stanu łączy też może powstać cykl w routingu. W tym celu skonstruuj sieć z dwoma wyróżnionymi, sąsiadującymi ze sobą routerami A i B. Załóż, że wszystkie routery znają graf całej sieci. W pewnym momencie łącze między A i B ulega awarii, o czym A i B od razu się dowiadują. Zalewają one sieć odpowiednią aktualizacją. Pokaż, że w okresie propagowania tej aktualizacji (kiedy dotarła ona już do części routerów a do części nie) może powstać cykl w routingu.



Krok 0:

—	A	B	C	D
A	0	1	2(B)	2 (B)
B	1	0	1	1
C	2(B)	1	0	1
D	2(B)	1	1	0

Krok 1: uszkodzenie

—	A	B	C	D
A	0	∞	2(B)	2 (B)

—	A	B	C	D
B	∞	0	1	1
C	2(B)	1	0	1
D	2(B)	1	1	0

Krok 2:

- A: “nie mam połączenia z B, aktualizuje swoją tablicę”
- C \rightarrow D “mam do A 2 kroki, przez B”
- B \rightarrow C “mam do A ścieżkę nieskończoną”

—	A	B	C	D
A	0	∞	∞	2 (B)
B	∞	0	1	1
C	∞	1	0	1
D	∞	1	1	0