

①

Sieci 1/

• 10.1.2.3/8

↑  
prefix

adres komputera w sieci: 10.0.0.0/8

alle której adres rozgłoszeniowy to: 10.255.255.255/8

• 156.17.0.0/16

inny adres: 10.0.0.1/8

bity poza prefixem to zero → to adres sieci

rozgłoszeniowy → 156.17.255.255/16

przydzielony komputer → 156.17.1.2/16

• 88.88.88.88/27

88 →  $\overset{36}{01100011}$

128 64 32 16 8 4 2 1

88.88.88.01100011/27

adres komputera w sieci: 88.88.88.88/27

broadcast: 88.88.88.127/27

inny adres: 88.88.88.100/27

• 156.17.64.4/30 010000010000 == 4

bity poza prefixem to zero → to adres sieci

broadcast: 156.17.64.7/30

inny adres: 156.17.64.5/30

• 123.123.123.123/32

prefix to cały adres (4 bajty, 32 bity),

węc jest to adres komputera (i jednocześnie sieci i broadcast)

2) 10.10.10.0/16

Strona 2/

10 → 0 0 0 0 1 0 1 0  
128 64 32 16 8 4 2 1

chcemy 5 podsieci (na 2 <sup>bitach</sup> ~~byłoby~~ zachodzący typ informacji o 4, potrzeba 3 bitów)

00 ... → 000  
01 ... → 001  
10 ... → 010  
11 ... → 100  
          → 110

128 64 32

10.10.0.0/18  
10.10.32.0/18  
10.10.64.0/18  
10.10.128.0/18  
10.10.192.0/18

Chcemy aby do adresowania 1 sieci potrzebowaliśmy 2  
spojonych adresów: sieci i rozszerzenia

Dla 5 podsieci mamy parę na każdy  $(2 \cdot 5) = 10$   
Czyli stworziliśmy  $10 - 2 = 8$  adresów dla komputerów.

Aby utworzyć najmniejszą podsieć, tak by były one rozdzielone i  
pomyślnie całą pulę 10.10.0.0/16 będziemy dzielić  
1 sieć na 2 inne.

fix 0 0 0 0 0 0 0 0 ① 10.10.0.0/16  
128 64 32 16 8 4 2 1 ② 10.10.0.0/17  
                                  10.10.128.0/17

③ 10.10.0.0/17  
10.10.128.0/18  
10.10.192.0/18

$$128 + 64 = 192$$

④ 10.10.0.0/17  
10.10.128.0/18  
10.10.192.0/18  
10.10.224.0/18

$$128 + 64 + 32 = 224$$

⑤ 10.10.0.0/17  
10.10.128.0/18  
10.10.192.0/18  
10.10.224.0/20  
10.10.240.0/20

10.10.240.0/20  
40.10.1111 0000.0/20

$2^{12} - 2 = 4094$  adresów  
dla komputerów

### Zadanie 3

Podsieć	Dokąd wysłać	Broadcast	
0.0.0.0/0	A	255.255.255.255/8	
10.0.0.0/23	B	10.0.1.255/23	
10.0.2.0/24	B	10.0.2.255/24	
10.0.3.0/24	B	10.0.3.255/24	
10.0.1.0/24	C	10.0.1.255/24	
10.0.0.128/25	B	10.0.0.255/25	$128 + 64 + 32 + 16 + 8 + 4 + 2 + 1$
10.0.1.8/29	B	10.0.1.15/29	$8 + 4 + 2 + 1 = 15$
10.0.1.16/29	B	10.0.1.23/29	$16 + 4 + 2 + 1 = 23$
10.0.1.24/29	B	10.0.1.31/29	"00011000", $24 + 4 + 2 + 1 = 31$

Widzimy że dla sieci 10.0.0.0/23, 10.0.2.0/24, 10.0.3.0/24, adresy są rozłączne i można je skleić (różnica na ostatnim bicie prefixu).

Skleimy do 10.0.0.0/22

Tak samo dla 10.0.1.16/29 oraz 10.0.1.24/29 skleiamy do 10.0.1.16/28

Podsieć	Dokąd wysłać	Broadcast
0.0.0.0/0	A	255.255.255.255/8
10.0.0.0/22	B	10.0.3.255/22
10.0.1.0/24	C	10.0.1.255/24
10.0.0.128/25	B	10.0.0.255/25
10.0.1.8/29	B	10.0.1.15/29
10.0.1.16/28	B	10.0.1.31/28

Widać też, że 10.0.0.128/25 to podzbiór 10.0.0.0/22 (jednocześnie rozłączny z siecią dla C!: 10.0.1.0/24)

Możemy go usunąć.

Podsieć	Dokąd wysłać	Broadcast
0.0.0.0/0	A	255.255.255.255/8
10.0.0.0/22	B	10.0.3.255/22
10.0.1.0/24	C	10.0.1.255/24
10.0.1.16/28	B	10.0.1.31/28
10.0.1.8/29	B	10.0.1.15/29

Dla ostatnich bajtów dwóch ostatnich sieci mamy "(0001) 0000" oraz "(0000 1)000" zatem są rozłączne (różny prefix) i nie da się ich połączyć nie zmieniając puli (np. 10.0.1.4 nie wpada do żadnej z nich)

## Zadanie 4

Podsieć	Dokąd wysłać	Broadcast		
0.0.0.0/0	A	255.255.255.255/8		
10.0.0.0/8	B	10.255.255.255/8		
10.3.0.0/24	C	10.3.0.255/24		
10.3.0.32/27	B	10.3.0.63/27	32 + 16 + 8 + 4 + 2 + 1 = 63	"0010 0000"
10.3.0.64/27	B	10.3.0.95/27	"+31"	"0100 0000"
10.3.0.96/27	B	10.3.0.127/27	"+31"	"0110 0000"

Dwa ostatnie wiersze sklejamy do: 10.3.0.64/26

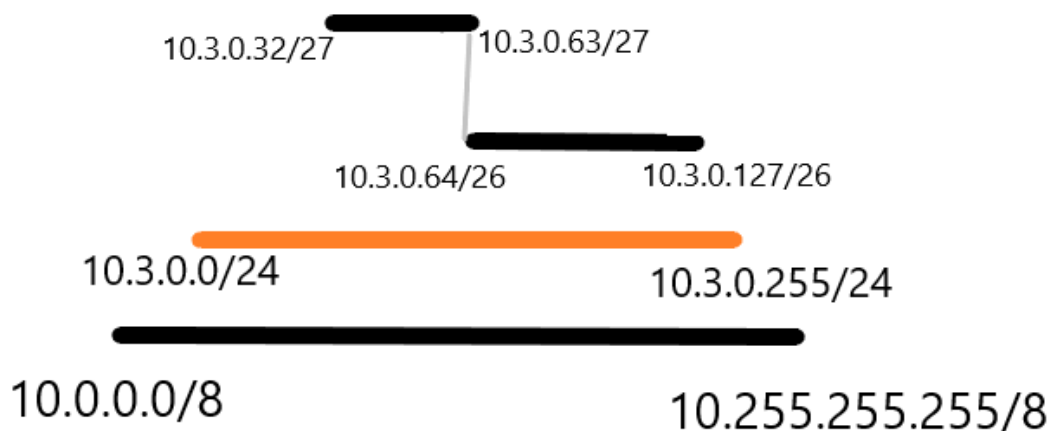
Podsieć	Dokąd wysłać	Broadcast
0.0.0.0/0	A	255.255.255.255/8
10.0.0.0/8	B	10.255.255.255/8
10.3.0.0/24	C	10.3.0.255/24
10.3.0.64/26	B	10.3.0.127/26
10.3.0.32/27	B	10.3.0.63/27

Wpis 3 zawiera się całkowicie w zakresach wpisu nr.2.

Natomiast 2 ostatnie zawierają się w zakresie wpisu nr.3 i są rozłączne. Nie da się ich łatwo skleić (różny rozmiar prefixu)

Ale skoro mamy  $B < C < B$  to gdyby podzielić wpis nr.3 na 2 części, to wpisy 4 i 5 były by redundantne bo te zakresy opisywałby wpis nr. 2

Rysunek:



Chcemy aby jeden zakres kończył się na 10.3.0.31 a drugi zaczynał od 10.3.0.128 i kończył na 10.3.0.255.

10.3.0.0/27 zeruje adresy do 10.3.0.31/27,

10.3.0.128/25 da ten zakres

Finalna tablica:

Podsiec	Dokąd wysłać	Broadcast
0.0.0.0/0	A	255.255.255.255/8
10.0.0.0/8	B	10.255.255.255/8
10.3.0.128/25	C	10.3.0.255/25
10.3.0.0/27	C	10.3.0.31/27



## Sieci komputerowe - zad 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.

Chcemy posortować od najdłuższego do najkrótszego prefiksu. Dla wpisów o tej samej długości prefiksu, kolejność nie ma znaczenia.

*Dowód.* Zakładamy że wpisy w tablicy routingu są posortowane malejąco względem długości prefiksu.

Chcemy pokazać, że pierwszy wybrany wpis jest tym najlepiej dopasowanym, gdzie dopasowanie wpisu oznacza, że jest zgodny na swoim prefiksie do adresu, a najlepsze dopasowanie oznacza takie dopasowanie, którego prefiks jest najdłuższy z dopasowań w całej tablicy routingu.

Weźmy dowolny adres IP  $x$ . Niech  $i$  oznacza pierwszy napotkany w tablicy routingu wpis dopasowany do  $x$ .

Oznacza to, że  $i$  oraz  $x$  jest zgodne na  $k$  bitach prefiksu.

Weźmy dowolny inny wpis  $j$  i załóżmy, że jest lepszym dopasowaniem, co oznacza, że jest on zgodny z  $x$  na  $l$  bitach gdzie  $k < l$

Skoro przechodzimy tablicę od początku do końca, to wiemy, że wpis  $j$  jest na wyższym indeksie.

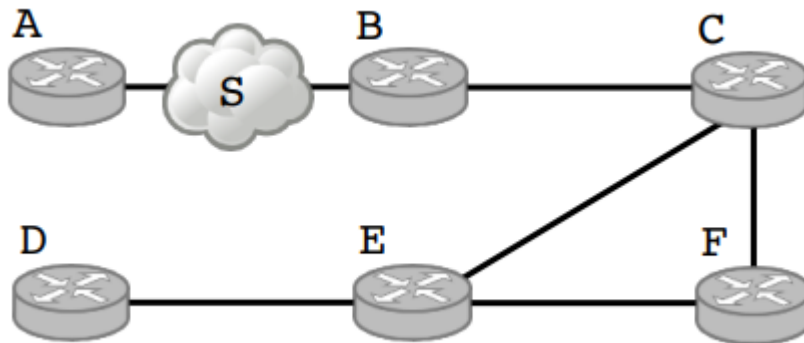
Z założenia, więc wpis  $j$  ma prefiks krótszy lub tej samej długości co wpis  $i$  ( $k \geq l$ ).

Mamy sprzeczność, zatem nie istnieje inny wpis  $j$  będący lepszym dopasowaniem.

□

## Zadanie 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?



Algorytm polega na wysyłaniu wektora odległości do swoich sąsiadów i aktualizowaniu na podstawie informacji od sąsiadów (minimalny dystans). Początkowo w wektorach routery mają tylko swoich sąsiadów.

0	A	B	C	D	E	F
trasa do A	-	1				
trasa do B	1	-	1			
trasa do C		1	-		1	1
trasa do D				-	1	
trasa do E			1	1	-	1
trasa do F			1		1	-
trasa do S	1	1				

1	A	B	C	D	E	F
trasa do A	-	1	2 (via B)			
trasa do B	1	-	1		2 (via C)	2 (via C)
trasa do C	2 (via B)	1	-	2 (via E)	1	1
trasa do D			2 (via E)	-	1	2 (via E)
trasa do E		2 (via C)	1	1	-	1
trasa do F		2 (via C)	1	2 (via E)	1	-
trasa do S	1	1	2 (via B)			

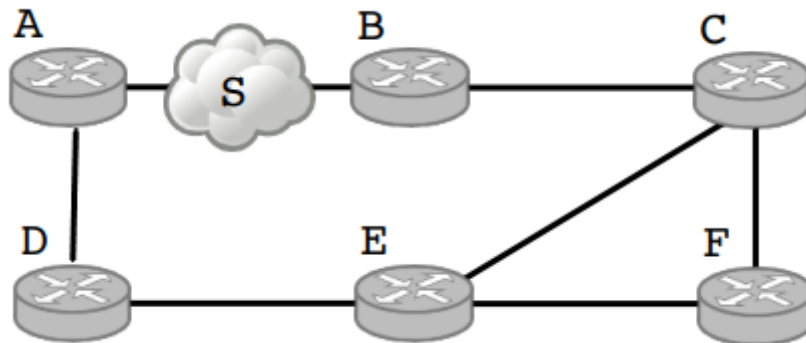
2	A	B	C	D	E	F
trasa do A	-	1	2 (via B)		3 (via C)	3 (via C)
trasa do B	1	-	1	3 (via E)	2 (via C)	2 (via C)
trasa do C	2 (via B)	1	-	2 (via E)	1	1
trasa do D		3 (via C)	2 (via E)	-	1	2 (via E)
trasa do E	3 (via B)	2 (via C)	1	1	-	1
trasa do F	3 (via B)	2 (via C)	1	2 (via E)	1	-
trasa do S	1	1	2 (via B)		3 (via C)	3 (via C)

3	A	B	C	D	E	F
trasa do A	-	1	2 (via B)	4 (via E)	3 (via C)	3 (via C)
trasa do B	1	-	1	3 (via E)	2 (via C)	2 (via C)
trasa do C	2 (via B)	1	-	2 (via E)	1	1
trasa do D	4 (via B)	3 (via C)	2 (via E)	-	1	2 (via E)
trasa do E	3 (via B)	2 (via C)	1	1	-	1
trasa do F	3 (via B)	2 (via C)	1	2 (via E)	1	-
trasa do S	1	1	2 (via B)	4 (via E)	3 (via C)	3 (via C)

Wystarczyły, więc 3 kroki, aby osiągnąć stabilność.



## Zadanie 7



obecna tabela (A i D są teraz w sąsiedztwie!)

0	A	B	C	D	E	F
trasa do A	-	1	2 (via B)	1	3 (via C)	3 (via C)
trasa do B	1	-	1	3 (via E)	2 (via C)	2 (via C)
trasa do C	2 (via B)	1	-	2 (via E)	1	1
trasa do D	1	3 (via C)	2 (via E)	-	1	2 (via E)
trasa do E	3 (via B)	2 (via C)	1	1	-	1
trasa do F	3 (via B)	2 (via C)	1	2 (via E)	1	-
trasa do S	1	1	2 (via B)	4 (via E)	3 (via C)	3 (via C)

skupiamy się na tym co A i D wyślą swoim sąsiadom (odpowiednio A wyśle B i D i D wyśle A i E)

1	A	B	C	D	E	F
trasa do A	-	1	2 (via B)	1	2 (via A)	3 (via C)
trasa do B	1	-	1	3 (via E)	2 (via C)	2 (via C)
trasa do C	2 (via B)	1	-	2 (via E)	1	1
trasa do D	1	2 (via A)	2 (via E)	-	1	2 (via E)
trasa do E	2 (via D)	2 (via C)	1	1	-	1
trasa do F	3 (via B)	2 (via C)	1	2 (via E)	1	-
trasa do S	1	1	2 (via B)	2 (via A)	3 (via C)	3 (via C)

i to jest już stan stabilny.