

Adam Kufel, nr indeksu: 292345, Sieci Komputerowe

Lista 1

Deklarowana liczba punktów za zadania: 7/10

Rozwiązania zadań: 1, 2, 3, 4, 5, 6, 7

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.

Adres sieci – najmniejszy możliwy adres w danej sieci. Ustawiamy bity części adresowej urządzenia na 0.

Adres rozgłoszeniowy – największy możliwy adres w danej sieci. Ustawiamy bity części adresowej urządzenia na 1.

• 10.1.2.3/8 – 0b00001010 xxxxxxxx xxxxxxxx xxxxxxxx

Jest to adres komputera.

Adres sieci: 10.0.0.0

Adres rozgłoszeniowy: 10.255.255.255

Adres dowolnego innego komputera w tej sieci: 10.21.31.4

• 156.17.0.0/16 – 0b10011100 00010001 xxxxxxxx xxxxxxxx

Jest to adres sieci.

Adres sieci: 156.17.0.0

Adres rozgłoszeniowy: 156.17.255.255

Adres dowolnego komputera w tej sieci: 156.17.10.5

• 99.99.99.99/27 – 0b01100011 01100011 01100011 011xxxxx

Jest to adres komputera.

Adres sieci: 99.99.99.(0b01100000) = 99.99.99.(64 + 32) = 99.99.99.96

Adres rozgłoszeniowy: 99.99.99.(0b01111111) = 99.99.99.(64 + 32 + 16 + 8 + 4 + 2 + 1) = 99.99.99.127

Adres dowolnego innego komputera w tej sieci: 99.99.99.102

• 156.17.64.4/30 – 0b10011100 00010001 01000000 000001xx

Jest to adres sieci.

Adres sieci: 156.17.64.(0b00000100) = 156.17.64.4

Adres rozgłoszeniowy: 156.17.64.(0b00000111) = 156.17.64.7

Adres dowolnego innego komputera w tej sieci: 156.17.64.5

• 123.123.123.123/32 – **Jest to jeden konkretny adres IP.**

Może być stosowany dla celów adresu pętli zwrotnej, wtedy taki adres jest wszystkim (adresem komputera, sieci, rozgłoszeniowym)

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?

Pula adresów w danej sieci: 10.10.0.0 – 10.10.255.255

Najpierw można podzielić całą sieć na 4 podsieci:

1 podsieć: 10.10.0.0/18 Adresy: 10.10.0.0 – 10.10.63.255
2 podsieć: 10.10.64.0/18 Adresy: 10.10.64.0 – 10.10.127.255
3. podsieć: 10.10.128.0/18 Adresy: 10.10.128.0 – 10.10.191.255
ostatnia podsieć: 10.10.192.0/18 Adresy: 10.10.192.0 – 10.10.255.255

A następnie jedną z podsieci można podzielić na dwie podsieci. Zamieniamy ostatnią podsieć na 2 podsieci:

4 podsieć 10.10.192.0/19 Adresy: 10.10.192.0 – 10.10.223.255
5 podsieć 10.10.224.0/19 Adresy: 10.10.224.0 – 10.10.255.255

Wynik:

- **1 podsieć: 10.10.0.0/18**
 - Adres sieci: 10.10.0.0
 - Adres rozgłoszeniowy: 10.10.63.255
- **2 podsieć: 10.10.64.0/18**
 - Adres sieci: 10.10.64.0
 - Adres rozgłoszeniowy: 10.10.127.255
- **3 podsieć: 10.10.128.0/18**
 - Adres sieci: 10.10.128.0
 - Adres rozgłoszeniowy: 10.10.191.255
- **4 podsieć 10.10.192.0/19**
 - Adres sieci: 10.10.192.0
 - Adres rozgłoszeniowy: 10.10.223.255
- **5 podsieć 10.10.224.0/19**
 - Adres sieci: 10.10.224.0
 - Adres rozgłoszeniowy: 10.10.255.255

Jak zmieniła się liczba adresów IP możliwych do użycia przy adresowaniu komputerów?

Pierwotnie w naszej sieci 10.10.0.0/16 mogliśmy zaadresować $2^{16} - 2$ komputerów (wypada nam adres sieci i adres rozgłoszeniowy). Dzieląc ją na 5 podsieci tracimy dodatkowo 4 adresy sieci i 4 adresy rozgłoszeniowe w 4 nowo utworzonych podsieciach. **Zatem po podziale możemy zaadresować $2^{16} - 10$ komputerów w całej podzielonej na podsieci sieci.**

Jaki jest minimalny rozmiar podsieci, który możemy uzyskać dzieląc naszą sieć na 5 podsieci?

Możemy rozważyć następującą strategię zachłanną: w każdej iteracji $i = 0, 1, 2, 3$ bierzemy największą możliwą podsieć tak, aby nie zużyć całej dostępnej puli adresowej sieci. W ostatniej iteracji, w której będziemy wybierać ostatnią podsieć, weźmy podsieć o rozmiarze podsieci wybranej w poprzedniej iteracji, aby dopełnić całą pulę adresową sieci i zakończyć podział.

Ponieważ w każdej iteracji braliśmy możliwie największe rozmiarowo podsieci, ostatnia i przedostatnia podsieć będzie podsiecią o minimalnym rozmiarze, jaki możemy osiągnąć dzieląc sieć na 5 podsieci:

Wynik podziału dla strategii zachłannej:

- **1 podsieć: 10.10.0.0/17**
 - Adres sieci: 10.10.0.0
 - Adres rozgłoszeniowy: 10.10.127.255
- **2 podsieć: 10.10.128.0/18**
 - Adres sieci: 10.10.128.0
 - Adres rozgłoszeniowy: 10.10.191.255
- **3 podsieć: 10.10.192.0/19**
 - Adres sieci: 10.10.192.0
 - Adres rozgłoszeniowy: 10.10.223.255
- **4 podsieć: 10.10.224.0/20**
 - Adres sieci: 10.10.224.0
 - Adres rozgłoszeniowy: 10.10.239.255
- **5 podsieć: 10.10.240.0/20**
 - Adres sieci: 10.10.240.0
 - Adres rozgłoszeniowy: 10.10.255.255

Odpowiedź: minimalny rozmiar podsieci jaki możemy uzyskać w naszym przypadku to 2^{12} - 2 puli adresowej dla komputerów przy masce podsieci /20

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.

Wyznaczmy zakresy obsługiwanych adresów dla każdego wpisu

1. A: 0.0.0.0/0 – wszystkie
2. B: 10.0.0.0/23 – od 10.0.0.0 do 10.0.1.255
3. B: 10.0.2.0/24 – od 10.0.2.0 do 10.0.2.255
4. B: 10.0.3.0/24 – od 10.0.3.0 do 10.0.3.255

5. C: 10.0.1.0/24 – od 10.0.1.0 do 10.0.1.255
6. B: 10.0.0.128/25 – od 10.0.0.128 do 10.0.0.255
7. B: 10.0.1.8/29 – 10.0.1.8 – 10.0.1.15
8. B: 10.0.1.16/29 – 10.0.1.16 - 10.0.1.23
9. B: 10.0.1.24/29 – 10.0.1.24 – 10.0.1.31

Zauważmy, że wpisy 2, 3, 4, 6 dla routera B obejmują ciągły zakres adresów od 10.0.0.0 do 10.0.3.255. Taki zakres możemy wyrazić adresem CIDR:

a) 10.0.0.0/22 – router B

Wewnątrz zakresu reguły a) znajduje się reguła dla routera C:

b) 10.0.1.0/24 – router C

Ponieważ adresy przychodzące są dopasowywane do zgodnej reguły o najdłuższym prefiksie sieci,

możemy stwierdzić, iż reguła b) zostanie zachowana poprawnie (ma dłuższy prefiks niż a))

Wpisy 7, 8, 9 możemy wyrazić za pomocą następującego adresu CIDR, z pewnym zastrzeżeniem:

c) 10.0.1.0/27 – router B

Otóż w regule c) znajduje się zakres adresów 10.0.1.0 – 10.0.1.7, które powinny trafić do routera C zgodnie z regułą b). Ponieważ prefiks reguły c) jest większy niż prefiks reguły b), tak się nie stanie.

Dlatego musimy dodać nową regułę o dłuższym prefiksie:

d) 10.0.1.0/29 – router C

Odpowiedź:

- **0.0.0.0/0** -> **do routera A**
- **10.0.0.0/22** -> **do routera B**
- **10.0.1.0/24** -> **do routera C**
- **10.0.1.0/27** -> **do routera B**
- **10.0.1.0/29** -> **do routera C**

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

Znowu wyznaczmy zakresy adresów:

1. A: 0.0.0.0/0 – wszystkie
2. B: 10.0.0.0/8 – od 10.0.0.0 do 10.255.255.255
3. C: 10.3.0.0/24 – od 10.3.0.0 do 10.3.0.255
4. B: 10.3.0.32/27 – od 10.3.0.32 do 10.3.0.63
5. B: 10.3.0.64/27 – od 10.3.0.64 do 10.3.0.95
6. B: 10.3.0.96/27 – od 10.3.0.96 do 10.3.0.127

Podobnie jak w poprzednim zadaniu zamieniamy reguły 4, 5, 6 na:

- a) 10.3.0.0/25 – router B (daje zakres od 10.3.0.0 do 10.3.0.127)

Z reguły a) musimy wyłączyć zakres od 10.3.0.0 do 10.3.0.31 należący do reguły 3:

- 3. 10.3.0.0/24 – router C

Osiągniemy to dodając nową regułę o dłuższym prefiksie od reguły a) obejmującą ten zakres:

- b) 10.3.0.0/27 – router C

Odpowiedź:

- **0.0.0.0/0** -> **do routera A**
- **10.0.0.0/8** -> **do routera B**
- **10.3.0.0/24** -> **do routera C**
- **10.3.0.0/25** -> **do routera B**
- **10.3.0.0/27** -> **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.

Sortujemy wpisy w tablicy routingu nierosnąco po długościach prefiksów sieci tak, aby wpisy od początku tablicy miały jak największe długości prefiksów sieci.

Wówczas wybór "pierwszy pasujący" dopasuje się do pierwszego pasującego wpisu, który będzie miał najdłuższy prefiks spośród wszystkich pasujących. Pozostałe pasujące wpisy będą znajdowały się za wybranym wpisem oraz będą miały mniejszy prefiks. Nie zaistnieje sytuacja, że będą dwa różne pasujące wpisy o tej samej długości prefiksu i tym samym prefiksie, gdyż z definicji sposobu adresowania CIDR.

Uzasadnienie formalne

Niech:

- X = zbiór danych wejściowych adresów IP
- ND = zasada najlepszego dopasowania
- PP = zasada pierwszy pasujący
- W = zbiór wpisów w tablicy routingu

Chcemy pokazać, że:

- Jeśli wpisy w tablicy routingu są posortowane nierosnąco po długościach prefiksów sieci, to:
 - (1) dla każdego x należącego do X $ND(x)$ zwraca taki sam wynik co $PP(x)$, czyli $ND(x) \Leftrightarrow PP(x)$

(Z): Załóżmy, że wpisy w tablicy routingu są posortowane nierosnąco po długościach prefiksów sieci.

Pokażemy, że zachodzi (1)

1. ' \Rightarrow '

Weźmy dowolny x należący do X . Wówczas ND wybierze, zgodnie z definicją reguły najlepszego dopasowania, ze zbioru wszystkich pasujących wpisów v z W o największym rozmiarze maski podsieci. PP natomiast wybierze pierwszy napotkany pasujący wpis w z W idąc od początku tablicy. Z (Z) wiemy, że wpisy w W są posortowane nierosnąco po długościach prefiksów sieci, więc wpis w będzie miał największą długość prefiksu spośród wszystkich pasujących wpisów. Dwa przypadki:

- Dla danej długości prefiksu $n \in N$ istnieje jeden pasujący wpis w w W . Wówczas $w == v$, gdyż zarówno PP i ND znajduje pasujący wpis o największej długości prefiksu w tej samej tablicy W .
- Dla danej długości prefiksu $n \in N$ istnieje więcej niż jeden pasujący wpis w W . Weźmy dwa takie wpisy $w1, w2$ z W i przyjrzyjmy się im. Ponieważ oba mają tę samą długość prefiksu oraz oba się dopasowały do prefiksu x , ich prefiksy są identyczne, zatem $w1 = w2$. Z tego znów wynika, że $w == v$.

Zakładamy poprawność tablicy routingu, tj. nie mogą istnieć dwa identyczne wpisy prowadzące do różnych routerów – wówczas żadna metoda nie ma pewności, który z nich jest poprawny.

2. ' \Leftarrow '

Weźmy dowolny x należący do X . Wówczas PP wybierze pierwszy napotkany pasujący wpis w z W o największym rozmiarze maski podsieci tak samo jak w 1. ' \Rightarrow '. Natomiast ND wybierze, zgodnie z definicją reguły najlepszego dopasowania, ze zbioru wszystkich pasujących wpisów v z W o największym rozmiarze maski podsieci. Dalsze rozważania są analogiczne jak w 1. ' \Rightarrow '.

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?

Tworzymy tabelę zgodnie z techniką podaną na wykładzie. Zgodnie z wykładem obiekt S posiada następującą interpretację:

- S reprezentuje sieć – koszt dojścia do S uwzględniamy, ale koszt przejścia przez S pomijamy. Zakładamy również, że w kolejnych krokach sąsiad połączony bezpośrednio prześle szybciej swoją tablicę routingu niż sąsiad połączony przez tę sieć.

Krok 0 – uzupełniamy tylko informacje o sąsiadach

	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				

Krok 1 – sąsiedzi przesyłają swoje tablice zgodne ze stanem w kroku 0. Tablica jest aktualizowana w przypadku otrzymania krótszej ścieżki (lub jakiegokolwiek ścieżki w przypadku pustego miejsca).

	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)			

Krok 2 – sąsiedzi przesyłają swoje tablice zgodne ze stanem w kroku 1. Aktualizacja następuje analogicznie jak w kroku 1.

	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)

Krok 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)

Dalsze kroki nie spowodują aktualizacji tablic. W 4 krokach (0, 1, 2, 3) został osiągnięty stan stabilny.

7. Załóżmy, że w powyższej sieci tablice routingu zostały już zbudowane. Co będzie się działo, jeśli zostanie dodane połączenie między routerami A i D?

Krok 0. Tablice routerów A i D zaktualizują się o informację o nowo powstałym sąsiedztwie

	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)

Krok 1. Pojawienie się nowego połączenia spowoduje aktualizację wybranych tras na bardziej korzystne trasy

	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	2 (via A)	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 następuje stan stabilny, gdyż w kroku 1 zaktualizowane trasy mają koszt 2, a najwyższy koszt w tablicach routerów to koszt 3, więc nowe trasy będą co najwyżej tak samo kosztowne, ale nie lepsze.