Sieci Komputerowe - Lista 1

Jakub Kuciński, prowadzący Andrzej Łukaszewski

Wrocław, Marzec 19, 2020

Spis treści

1 Zadanie 1

Kropki w numerze IP rozdzielają bajty danego adresu IP. Adres rozgłoszeniowy jest ostatnim adresem, adres sieci to pierwszy adres, a adresy komputerów to wszystkie pomiędzy nimi. Liczba po / oznacza długość prefiksu sieci (w bitach).

- 10.1.2.3/8 : podany adres to adres komputera, adres sieci - 10.0.0.0, adres rozgłoszeniowy 10.255.255.255, adres jakiegoś komputera - 10.0.0.1

- 156.17.0.0/16 : podany adres to adres sieci, adres sieci - 156.17.0.0/16, adres rozgłoszeniowy - 156.17.255.255, adres jakiegoś komputera - 156.17.11.11

- 99.99.99.99/27 = 01100011.01100011.01100011.01100011/27 : podany adres to adres komputera, adres sieci - 99.99.99.96, adres rozgłoszeniowy - 99.99.99.127, adres jakiegoś komputera - 99.99.99.100

- 156.17.64.4/30 : podany adres to adres sieci, adres sieci - 156.17.64.4, adres rozgłoszeniowy - 156.17.64.7, adres jakiegoś komputera - 156.17.64.5

- 123.123.123.123/32 : jest to sieć z jednym adresem IP (jeden komputer), w tym przypadku adres rozgłoszeniowy, sieci i komputera są podanym adresem IP

2 Zadanie 2

Adres sieci 10.10.0.0/16 z zadania zapisany binarnie: 00001010.00001010.00000000.00000000 i jej maska 11111111.11111111.00000000.00000000. Możemy podzielić tę sieć na pół. Pierwszą połowę przeznaczymy na pierwszą sieć, a drugą podzielimy na kolejno 4 równoliczne rozłączne sieci:

- Adres pierwszej (większej) sieci - 00001010.00001010.00000000.00000000, maska - 11111111.11111111.10000000.00000000. Zamieniając na zapis dziesiętny - 10.10.0.0/17
- Adres drugiej sieci - 00001010.00001010.10000000.00000000, maska - 11111111.11111111.11100000.00000000. Zamieniając na zapis dziesiętny - 10.10.128.0/19
- Adres trzeciej sieci – 00001010.00001010.10100000.00000000, maska - 11111111.11111111.11100000.00000000. Zamieniając na zapis dziesiętny - 10.10.160.0/19
- Adres czwartej sieci – 00001010.00001010.11000000.00000000, maska - 11111111.11111111.11100000.00000000. Zamieniając na zapis dziesiętny - 10.10.192.0/19
- Adres piątej sieci - - 00001010.00001010.10100000.00000000, maska - 11111111.11111111.11100000.00000000. Zamieniając na zapis dziesiętny - 10.10.224.0/19

Każda z tych podsieci zawiera dwa adresy, które nie mogą być przypisane komputerom - adres sieci i adres rozgłoszeniowy. Skoro mamy 5 podsieci to łącznie 10 adresów nie może zostać wykorzystanych do adresowania komputerów. Natomiast oryginalna sieć 10.10.0.0/16 zawierała już adres rozgłoszeniowy i adres sieci, zatem liczba niemożliwych do przypisania adresów wzrosła o 8.

Jaka jest najmniejsza możliwa sieć? Powiedzmy, że ma prefiks długości k. Nasze 5 podsieci jest rozłącznych i wypełniają całą sieć 10.10.0.0/16, zatem musi istnieć inna podsieć, która ma prefiks takiej samej długości jak k, bo musi ją "uzupełnić" do większej sieci. Razem tworzą sieć o prefiksie długości k-1. Ona z kolei znowu musi być uzupełniona przez inną sieć o prefiksie k-1. Razem tworzą sieć o prefiksie k-2. Znowu uzupełniamy, łączymy i uzupełniamy. Mamy sieć o prefiksie k-4 i wykorzystaliśmy do tego już wszystkie dostępne 5 podsieci, zatem musieliśmy dostać sieć 10.10.0.0/16, czyli $k - 4 = 16$, stąd $k = 20$. Zatem najmniejsza sieć ma $2^{32-20} - 2 = 4094$ adresów dostępnych dla komputerów.

3 Zadanie 3

1) 0.0.0.0/0 → do routera A
2) 10.0.0.0/23 → do routera B
3) 10.0.2.0/24 → do routera B
4) 10.0.3.0/24 → do routera B
5) 10.0.1.0/24 → do routera C

6) 10.0.0.128/25 → do routera B

7) 10.0.1.8/29 → do routera B

8) 10.0.1.16/29 → do routera B

9) 10.0.1.24/29 → do routera B

Wiemy, że w tablicy routingu wybierana jest reguła z najdłuższym prefiksem. Jeśli miniemy wpis 5), to możemy wpaść do routera B albo do ogólnego A. Zauważmy, że wtedy możemy zapisać 2), 3) i 4) łącznie jako 10.0.0.0/22 → do routera B. Widzimy, że wpisy 5)-9) są uszczegółowieniami 10.0.0.0/22 oraz występują w nich tylko routery B i C. Możemy zauważyć, że wpis 6) jest rozłączny z jedynym wpisem z routerem C, czyli 5), więc skoro uszczegóławia 10.0.0.0/22 → do routera B to możemy go pominąć (bo też prowadzi do B). Wpisy 7)-9) tworzą zwartą część, która uszczegóławia wpis 5). Możemy zatem zastąpić je dwoma wpisami. Mniej szczegółowym 10.0.1.0/27 → do routera B oraz bardziej szczegółowym 10.0.1.0/29 → do routera C. W ten sposób otrzymujemy tablicę routingu:

- 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 Zadanie 4

1) 0.0.0.0/0 → do routera A

2) 10.0.0.0/8 → do routera B

3) 10.3.0.0/24 → do routera C

4) 10.3.0.32/27 → do routera B

5) 10.3.0.64/27 → do routera B

6) 10.3.0.96/27 → do routera B

Analogicznie jak w zadaniu 3. Widzimy, że wpisy 3)-6) są uszczegółowieniami 10.0.0.0/8 oraz występują w nich tylko routery B i C. Wpisy 4)-6) tworzą zwartą część, zawartą we wpisie 3). Nieprzysłonięty przez wpisy 4)-6) fragment wpisu 3) można rozbić na dwie części 10.3.0.128/25 oraz 10.3.0.0/27. Nie będą one miały żadnego wspólnego adresu z wpisami 4)-6), a skoro 4)-6) leżą wewnątrz 2) to będzie można je pominąć. W ten sposób otrzymamy:

- 0.0.0.0/0 → do routera A
- 10.0.0.0/8 → do routera B
- 10.3.0.128/25 → do routera C
- 10.3.0.0/27 → do routera C

5 Zadanie 5

Teza: uporządkowanie wpisów w tablicy routingu według długości prefiksu od najdłuższego do najkrótszego i wybór pierwszego pasującego jest równoważne braniu najlepszego dopasowania (najdłuższy pasujący prefiks).

Dowód:
Załóżmy nie wprost, że wybór pierwszego pasującego z naszej uporządkowanej tablicy routingu nie jest równoważny braniu najlepszego dopasowania. Weźmy adres ip, dla którego teza nie zachodzi i nazwijmy go Addr. Niech WpisUporz oznacza pierwszy wpis pasujący do Addr w uporządkowanej tablicy oraz WpisNajl oznacza wpis, który jest najlepszym dopasowaniem (najdłuższy prefiks). Ponadto przez PreUporz oznaczmy długość prefiksu WpisUporz oraz przez PreNajl oznaczmy długość prefiksu WpisNajl. Rozważmy przypadki:

- PreUporz > PreNajl
  Długość prefisku WpisNajl jest z założenia największa. Ale założyliśmy, że PreUporz > PreNajl. Sprzeczność.

- PreUporz = PreNajl
  Wiemy, że PreNajl to największa długość prefiksu wpisu pasującego do Addr, a skoro PreUporz = PreNajl, to WpisUporz też jest najbardziej pasującym wpisem. Sprzeczność (bo PreUporz miał być nienajlepszy).

- PreUporz < PreNajl
  Wiemy, że WpisNajl znajduje się w tablicy. Zastanówmy się gdzie znajduje sie w tablicy uporządkowanej. Skoro ma dłuższy prefiks to znajduje się przed WpisUporz, ale skoro pasuje do Addr i znajduje się przed WpisUporz, a bierzemy pierwszy pasujący wpis, to powinniśmy wziąć WpisNajl, a nie WpisUporz. Sprzeczność.

Z powyższych przypadków dochodzimy do sprzeczności, a stąd teza była prawdziwa.

6 Zadanie 6

Początkowy stan tablicy routingu:

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

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

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

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

Jak widać stan stabilny zostanie osiągnięty po 3 krokach

7 Zadanie 7

Tablica routingu po dodaniu połączenia między routerami A i D.

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

Uaktualniona tablica routingu:

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

8 Zadanie 8

Trasa do E po awarii łącza między D a E (bez straty ogólności przymujemy, że ścieżka z A do E prowadziła przez B a nie C):

	A	B	C	D
do E	3 (via B)	2 (via D)	2 (via D)	∞

D wysyła informacje do B i C o swoim sąsiedztwie. Interesuje nas to, że jego ścieżka do E wynosi ∞. Skoro droga z B i C prowadziła do E przez D to nowe drogi wynoszą teraz ∞.

	A	B	C	D
do E	3 (via B)	∞	∞	∞

Następnie A wysyła do B zatrutą ścieżkę (bo droga do E prowadzi przez B) oraz informuje C, że ma ścieżkę długości 3 do E. Droga z C do E przez A jest krótsza od ∞, więc zostaje wpisana do C.

	A	B	C	D
do E	3 (via B)	∞	4 (via A)	∞

C wysyła do A zatrutą ścieżkę oraz informuje D, że ma ścieżkę do E przez A. D uaktualnia swoją ścieżkę do E.

	A	B	C	D
do E	3 (via B)	∞	4 (via A)	5 (via C)

Teraz D wysyła informacje do B, że ma ścieżkę do E przez C. B uaktualnia swoją ścieżkę do E.

	A	B	C	D
do E	3 (via B)	6 (via D)	4 (via A)	5 (via C)

Otrzymaliśmy cykl (A,B,D,C).

9 Zadanie 9

10 Zadanie 10

Powiedzmy, że mamy pewną sieć do której należą połączone routery A i B z drogą od A do B. Zauważmy, że jeśli w wyniku wysłania komunikatu przez pewien początkowy router, do routera A dotrze $2^{\Omega(n)}$ komunikatów, to przesłanie ich wszystkich do B zajmie nam $2^{\Omega(n)}$ kroków (bo możemy wysłać tylko jeden komunikat naraz). Czyli rozesłanie tej informacji po całej sieci zajmie przynajmniej $2^{\Omega(n)}$ kroków. Skonstruujmy taką sieć. Lista kroków konstrukcji:

S $\leftarrow$ zbiór routerów należących do sieci

k $\leftarrow$ 0
Dopóki $i \leqslant n$:

- k $\leftarrow$ k+1
- Utwórz router o numerze k
- Dla każdego j $\in$ S dodaj połączenie z j do k
- Dodaj k do S

Skonstruowana sieć ma tą własność, że dla j-tego routera wszystkie połączenia wychodzące z j do k spełniają własność $k \in [j+1, n]$. Z kolei wszystkie połączenia wchodzące z k do j spełaniają własność $k \in [1, j-1]$. Pierwsze komunikaty zostają wysłane przez router 1. Oznaczmy przez S_k wynik algorytmu po k-tej iteracji. Zauważmy, że liczba komunikatów $f_1$dla S_1, która dojdzie do routera 1 wynosi 0. Liczba komunikatów f_2 dla S_2 oraz 2 wynosi 1. Dla S_3 liczba komuniaktów które dostanie 3 jest równa liczbie komunikatów, które dochodzą do 1 oraz 2 (zostaną przekazane dalej do 3) plus komunikat od 1, czyli $f_3 = 1 + f_1 + f_2$. Dalej dostaniemy $f_4 = 1 + f_1 + f_2 + f_3$, ... Możemy stąd wyznaczyć wzór $f_n = 1 + \sum_{i=0}^{n-1} f_i$. Przekształćmy wzór:
$f_n = 1 + \sum_{i=0}^{n-1} f_i = 1 + \sum_{i=0}^{n-2} f_i + f_{n-1} = (1 + \sum_{i=0}^{n-2} f_i) + f_{n-1} = f_{n-1} + f_{n-1} = 2 \cdot f_{n-1}$.
Zauważając, że $f_2 = 1$ dostajemy $f_n = 2^{n-2}$. Czyli do n-tego routera trafia w naszej sieci 2^{n-2} komunikatów. Potrzebujemy jednego oddzielnego routera, do którego nasz n-ty router będzie przesyłał te komunikaty, więc zabierając zmniejszając liczbę routerów w S o 1 dostajemy 2^{n-3} komunikatów dochodzących do n-tego routera. Przesłanie tych komunikatów do tego oddzielnego wyróżnionego zajmie 2^{n-3}. Wysyłanie tych komunikatów zacznie się, gdy dostaniemy pierwszy komunikat od routera 1 i będzie trwało to aż do wysłania ostatniego z kolejki. Oczywiście ta operacja będzie wykonywała się najdłużej ze wszystkich routerów, stąd przesłanie informacji zakończy się po czasie $2^{\Omega(n)}$.