

MATERIAŁY DYDAKTYCZNE: System binarny w adresacji IP i podstawy podziału na podsieci

1. Dlaczego komputer potrzebuje systemu binarnego?

W poprzednim temacie dowiedzieliśmy się, że adres IPv4 składa się z 32 bitów podzielonych na 4 oktety. Choć dla człowieka zapisujemy go dziesiętnie (np. 192.168.1.1), routery i przełączniki sieciowe przetwarzają te dane wyłącznie w **systemie binarnym (dwójkowym)**, czyli jako ciągi zer (0) i jedynek (1). Stan „1” oznacza obecność sygnału elektrycznego lub świetlnego, a „0” jego brak.

Aby sprawnie konfigurować sieci, dzielić je na mniejsze segmenty (tzw. subnetting) oraz rozumieć działanie masek podsieci, każdy informatyk musi opanować szybką konwersję między systemem dziesiętnym a binarnym w zakresie jednego oktetu (od 0 do 255).

2. Metoda Wagowa (Pozycyjna) – Klucz do konwersji

Jeden oktet to dokładnie 8 bitów. Każda pozycja bitu w okciecie ma przypisaną swoją stałą wartość (wagę), która jest kolejną potęgą liczby 2, licząc od prawej strony (od 2^0 do 2^7).

Pozycja bitu (od lewej)	1	2	3	4	5	6	7	8
Zapis potęgowy	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Waga (Wartość)	128	64	32	16	8	4	2	1

Zamiana z systemu binarnego na dziesiętny:

Jeśli podany bit wynosi 1, dodajemy jego wagę. Jeśli wynosi 0, pomijamy go.

Przykład: Zamień ciąg binarny 10100100 na liczbę dziesiętną.

Rozwiązanie:

$$(1 * 128) + (0 * 64) + (1 * 32) + (0 * 16) + (0 * 8) + (1 * 4) + (0 * 2) + (0 * 1) = 128 + 32 + 4 = 164$$

Zamiana z systemu dziesiętnego na binarny:

Idąc od lewej strony (od wagi 128), sprawdzamy, czy nasza liczba jest większa bądź równa wadze. Jeśli tak – wpisujemy 1 i odejmujemy wagę od liczby. Jeśli nie – wpisujemy 0 i przechodzimy do kolejnej wagi.

Przykład: Zamień liczbę 75 na postać binarną.

- Czy 75 >= 128? Nie -> 0
- Czy 75 >= 64? Tak -> 1 (75 - 64 = 11 zostaje)
- Czy 11 >= 32? Nie -> 0
- Czy 11 >= 16? Nie -> 0
- Czy 11 >= 8? Tak -> 1 (11 - 8 = 3 zostaje)
- Czy 3 >= 4? Nie -> 0
- Czy 3 >= 2? Tak -> 1 (3 - 2 = 1 zostaje)
- Czy 1 >= 1? Tak -> 1 (1 - 1 = 0)

Wynik dla liczby 75: 01001011

3. Wprowadzenie do podziału na podsieci (Subnetting)

Podział na podsieci (Subnetting) polega na podziale jednej dużej sieci fizycznej na mniejsze, logiczne segmenty. Domyślne klasy adresów (A, B, C) marnowały ogromną pulę adresów (np. klasa C oferuje zawsze 254 hosty – jeśli firma ma tylko 10 komputerów, 244 adresy przepadają).

Dlaczego dzielimy sieci?

- **Bezpieczeństwo:** Możemy odizolować komputery uczniów od komputerów dyrekcji i księgowości.
- **Wydajność:** Zmniejszenie ruchu rozgłoszeniowego (Broadcast). Pakiety rozgłoszeniowe nie krążą po całej szkole, a jedynie wewnątrz jednej pracowni.
- **Oszczędność:** Lepsze i bardziej precyzyjne zagospodarowanie dostępnych adresów IP.

4. Jak działa maska podsieci w praktyce?

Maska podsieci składa się z nieprzerwanego ciągu jedynek, po którym następuje ciąg zer.

Zmieniając domyślną maskę podsieci, dokonujemy tzw. „pożyczania” bitów z części hosta na rzecz części sieciowej.

Spójrzmy, jak tworzą się typowe wartości oktetów maski podsieci poprzez zapełnianie bitów

jedynkami od lewej strony:

- $10000000 = 128$
- $11000000 = 128 + 64 = 192$
- $11100000 = 128 + 64 + 32 = 224$
- $11110000 = 128 + 64 + 32 + 16 = 240$
- $11111000 = 128 + 64 + 32 + 16 + 8 = 248$
- $11111100 = 128 + 64 + 32 + 16 + 8 + 4 = 252$
- $11111111 = 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255$

Stąd właśnie biorą się maski takie jak 255.255.255.192 czy 255.255.255.240. Wartości końcowe masek nigdy nie są przypadkowe!

5. Formuły matematyczne sieci

Do wyliczania liczby podsieci oraz liczby dostępnych hostów stosujemy dwa proste wzory:

1. **Liczba utworzonych podsieci = 2^n** (gdzie n to liczba pożyczonych bitów ze strefy hosta).
2. **Liczba hostów w każdej podsieci = $2^h - 2$** (gdzie h to liczba bitów, które pozostały w strefie hosta jako zera). Odejmujemy 2, ponieważ pierwsza wartość to adres sieci, a ostatnia to adres rozgłoszeniowy.

KARTA PRACY UCZNIA: System binarny i podstawy subnettingu

Imię i nazwisko: Klasa: Data:

Zadanie 1. Konwersja binarno-dziesiętna (Użyj metody wagowej)

a) Zamień podane liczby dziesiętne na postać binarną (zapisz pełne 8 bitów):

- $12 = \dots\dots\dots$
- $192 = \dots\dots\dots$
- $105 = \dots\dots\dots$
- $254 = \dots\dots\dots$

b) Zamień podane ciągi binarne na liczby dziesiętne:

- $00001111 = \dots\dots\dots$
- $10000001 = \dots\dots\dots$
- $11110000 = \dots\dots\dots$
- $10101010 = \dots\dots\dots$

Zadanie 2. Analiza maski podsieci

Rozpisz podaną maskę podsieci w postaci binarnej i wskaż, ile bitów przeznaczono na sieć (jedyne), a ile na hosty (zera).

Maska: 255.255.255.192

- Zapis binarny maski:
- Łączna liczba bitów sieci (jedyne): (Notacja CIDR: /.....)
- Łączna liczba bitów hosta (zera):

Zadanie 3. Wyznaczanie liczby hostów (Analiza problemu)

Administrator sieci w Zespole Szkół Politechnicznych zmienił maskę podsieci w pracowni na maskę /26 (czyli 255.255.255.192). Oznacza to, że w ostatnim oktecie na część hosta pozostało dokładnie **6 bitów mierzonych jako zera** (32 bity - 26 bitów sieci = 6 bitów hosta).

Korzystając ze wzoru ($2^n - 2$), oblicz ilu maksymalnie użytkowników (hostów) można podłączyć do tak skonfigurowanej podsieci. Pokaż swoje obliczenia.

Obliczenia:

Odpowiedź:

.....