Zamiana liczba zapisanych w dowolnym systemie na system dziesiętny:

W systemie pozycyjnym o podstawie 10 wartości kolejnych cyfr odpowiadają kolejnym potęgom liczby 10 licząc od strony prawej i numerując od 0. Do dyspozycji mamy dziesięć cyfr: od 0 do 9.

Przykład:
$1259_{(10)} = 1 * 10^3 + 2 * 10^2 + 5 * 10^1 + 9 * 10_0 = 1 * 1000 + 2 * 100 + 5 * 10 + 9 * 1$

W systemie pozycyjnym o podstawie 2 (system binarny) wartości kolejnych cyfr odpowiadają kolejnym potęgom liczby 2 licząc od strony prawej i numerując od 0. Do dyspozycji mamy dwie cyfry: 0 i 1.
Potęgi liczby 2 tworzą następujący ciąg (dla 8 bitów, od pozycji najbardziej znaczącej):

128	64	32	16	8	4	2	1

Przykład:
$110111_{(2)} = 1 * 2^5 + 1 * 2^4 + 0 * 2^3 + 1 * 2^2 + 1 * 2^1 + 1 * {}^{20} = 32 + 16 + 4 + 2 + 1 = 55_{(10)}$

W systemie pozycyjnym o podstawie 16 (system szesnastkowy) wartości kolejnych cyfr odpowiadają kolejnym potęgom liczby 16 licząc od strony prawej i numerując od 0. Do dyspozycji mamy szesnaście cyfr: od 0 do 9 oraz A, B, C, D, E i F. Litery reprezentują odpowiednio wartości od 10 do 15.

Przykład:
$AB1_{(16)} = 10 * 16^2 + 11 * 16^1 + 1 * 16^0 = 2560 + 176 + 1 = 2737_{(10)}$

Zamiana liczb zapisanych w systemie dziesiętnym na system szesnastkowy i binarny

Aby zamienić liczbę zapisaną w systemie dziesiętnym na dowolny inny system liczbowy należy dzielić wskazaną liczbę przez podstawę systemu, na który dokonujemy zamiany aż do uzyskania zera.
Dla każdego kroku reszta z dzielenia wynosząca to wartość, którą umieszczamy na danej pozycji.
Oznaczenia:
/ – dzielenie całkowite, % – reszta z dzielenia (tzw. operacja **modulo**).

Przykład:
Zamiana (10) → (2)

25	25 / 2 = 12	25 % 2 = 1
12	12 / 2 = 6	12 % 2 = 0
6	6 / 2 = 3	6 % 2 = 0
3	3 / 2 = 1	3 % 2 = 1
1	1 / 2 = 0	1 % 2 = 1

Uzyskane wartości zapisujemy od ostatniej do pierwszej reszty z dzielenia:
$25_{(10)} = 11001_{(2)}$

Zamiana (10) → (16)

175	175 / 16 = 10	175 % 16 = 15
10	10 / 16 = 0	10 % 16 = 10

Uzyskane wartości zapisujemy od ostatniej do pierwszej reszty z dzielenia pamiętając o zastąpieniu wartości większych niż 9 odpowiadającymi im literami (w przykładzie: 10 – A, 15 – F):
$175_{(10)} = AF_{(16)}$

Zadanie
Dokonaj konwersji poniższych liczb pomiędzy wybranymi systemami liczbowymi;

$231_{(10)}$ = ……………………….$_{(2)}$ $01110111_{(2)}$ =………………………. $_{(10)}$ $85_{(16)}$ =………………………. $_{(2)}$

$237_{(10)}$ = ………………………. $_{(16)}$ $11011011_{(2)}$ =………………………. $_{(16)}$ $97_{(16)}$ = ………………………. $_{(10)}$

Zamiana liczb pomiędzy systemem binarnym a szesnastkowym

W systemie binarnym każde 4 cyfry (licząc od strony prawej) odpowiadają jednej cyfrze systemie szesnastkowym. Stąd można dokonać prostej zamiany:

$A1B5_{(16)}$ = 1010 0001 1011 $0101_{(2)}$, gdyż:

Wartości w systemie

(16)	A	1	B	5
(10)	10	1	11	5
(2)	1010	0001	1011	0101

Zadanie

Dokonaj konwersji poniższych liczb pomiędzy wybranymi systemami liczbowymi:

$AB_{(16)}$ = ……………………….$_{(2)}$ 0111 $0111_{(2)}$ =………………………. $_{(16)}$

$23_{(16)}$ = ………………………. $_{(2)}$ 1101 $1011_{(2)}$ =………………………. $_{(16)}$

Reprezentacja binarna liczb ujemnych

Aby zapisać liczby ujemne w systemie binarnym w komputerze stosuje się kod uzupełnień do dwóch. W celu wyodrębnienia liczb ujemnych pierwszy bit liczby binarnej stanowi o jej znaku: ‘0’ oznacza ‘+’, ‘1’ oznacza ‘–‘. Ważne jest przy tym ustalenie ilości bitów przeznaczonych na zapis liczby.

Dla dowolnej dodatniej liczby X zapisanej w systemie dziesiętnym zamiana na kod uzupełnień do dwóch jest identyczna jak zamiana tej liczby na system binarny.
Dla dowolnej ujemnej liczby X zapisanej w systemie dziesiętnym zamiana na kod uzupełnień do dwóch wygląda następująco:
$X_{(10)} = 2^N - | X_{(10)} | = Y_{(10)}$,
$Y_{(10)} \rightarrow A_{(U2)}$,
gdzie N jest liczbą bitów przeznaczonych na zapis liczby.

Przykład:
Mając do dyspozycji 8 bitów zamień wartość $-25_{(10)}$ na kod uzupełnień do 2:
$-25_{(10)} = 2^8 - 25 = 256 - 25 = 231$
$231_{(10)} = 1110\ 0111_{(2)}$
Więc: $-25_{(10)} = 1110\ 0111_{(U2)}$

Zadanie
Zapisz liczby dziesiętne w systemie uzupełnień U2:

$-20_{(10)}$ = …………………….$_{(U2)}$

$-230_{(10)}$ = ……………………. $_{(U2)}$

Zamiana liczb zapisanych w kodzie uzupełnień do dwóch na system dziesiętny

Dla dowolnej dodatniej liczby X zapisanej w kodzie uzupełnień do dwóch zamiana na system dziesiętny jest identyczna jak zamiana tej liczby z systemu binarnego na system dziesiętny.
Dla dowolnej ujemnej liczby X zapisanej w kodzie uzupełnień do dwóch zamiana na system dziesiętny wygląda następująco:
$X_{(U2)} \rightarrow Y_{(10)}$,
$Y_{(10)} - 2_N = A_{(10)}$,
gdzie N jest liczbą bitów przeznaczonych na zapis liczby.

Przykład:
Mając do dyspozycji 8 bitów zamień liczbę $11100111_{(U2)}$ na jej reprezentację w systemie dziesiętnym:
$1110\ 0111_{(U2)} = 231_{(10)}$

231 $_{(10)}$ – 2_8 = 231 – 256 = -25
Więc: 1110 0111$_{(U2)}$ = -25 $_{(10)}$

Zadanie
Przedstaw poniższe liczby w kodzie U2 w systemie dziesiętnym:

a) 10011110 ……………………………..

b) 01001011 ……………………………..

Liczba przeciwna w U2

Aby zamienić liczbę w U2 na przeciwną, należy wykonać dwa kroki:

* dokonać inwersji bitów, czyli pozamieniać 0 na 1 i odwrotnie;
* zwiększyć wynik o 1.

Przykład
Dana jest liczba: 01001010 $_{(U2)}$ = 74 $_{(10)}$

Dokonujemy inwersji: 10110101

Zwiększamy o 1:
10110110 $_{(U2)}$ = $1 * (-)2^7 + 0 * 2^6 + 1 * 2^5 + 1 * 2^4 + 0 * 2^3 + 1 * 2^2 + 1 * 2^1 + 0 * 2^0 +$ = -74$_{(10)}$

Zadanie
Zapisz liczby przeciwne do podanych w U2:

a) 10011110 ……………………………..

b) 01001011 ……………………………..

Dodawanie liczb binarnych

W celu dodania dwu liczb dwójkowych P i Q dokonuje się sumowania **par bitów** na poszczególnych pozycjach, rozpoczynając od najmniej znaczącego bitu. Podobnie jak w arytmetyce dziesiętnej należy przy sumowaniu bitów na każdej (i-tej) pozycji uwzględnić bit przeniesienia.

Przykład:
P = $1001_{(2)}$ = $9_{(10)}$
Q = $1101_{(2)}$ = $13_{(10)}$

wynik Y = 10110 L(Y) = 22

Liczby dwójkowe mogą być dodawane tylko parami, tzn., nie można dodawać „słupków" utworzonych z liczb dwójkowych. Dodawanie większej liczby argumentów dwójkowych wykonuje się zazwyczaj przez sekwencję dodawania poszczególnych argumentów. Na przykład, w celu dodania liczb A, B, C, D dodaje się najpierw A+B=W1, a następnie W1+C=W2 i W2+D=W. Taki proces sumowania kolejnych argumentów określa się jako sumowanie akumulacyjne.

Zadanie
Przeprowadź binarną operację dodawania liczb P + Q dla P = $1001_{(2)}$ Q = $1101_{(2)}$

Wynik: ……………………………..

Odejmowanie

W dwójkowym systemie liczbowym nie ma operacji odejmowania analogicznej do tej, która jest powszechnie używana w dziesiętnym systemie liczbowym. W związku z tym również układy cyfrowe nie mogą wykonywać takiej operacji odejmowania - mogą tylko dodawać. Dlatego, aby wykonać operację odejmowania liczb dwójkowych, należy dokonać zabiegu, który umożliwi wykorzystanie do tego celu operacji dodawania.
Zabiegiem takim jest przedstawienie odjemnika w kodzie uzupełnieniowych i następnie wykonanie operacji dodawania z odjemna. Czyli:
należy przekształcić odjemną do U2
powstał odjemnik, który należy dodać do odjemnej, na końcu odtrącić pierwszy bit.

Przykład:
$00011101_{(2)}$ – $00001001_{(2)}$ = ?

$00011101_{(2)}$ -> $29_{(10)}$
$00001001_{(2)}$ -> $9_{(10)}$ -> 11110110 + 00000001 = 11110111 $_{(U2)}$

$00011101_{(2)}$ +
11110111 $_{(U2)}$ =

00010100 -> $20_{(10)}$

Zadanie

Przeprowadź binarną operację odejmowanie liczb P - Q dla P = $0101_{(2)}$ Q = $1001_{(2)}$

Wynik: ……………………………..

Zadanie

Przeprowadź binarną operację odejmowanie liczb P - Q dla P = $010101_{(2)}$ Q = $111001_{(2)}$

Wynik: ……………………………..