Informatica A – Esercitazione – 19 settembre 2024

Codifica Binaria

1. *Da decimale a binario*

Per convertire un numero dal sistema binario al sistema decimale si deve dividere per 2 il numero, tenere il resto, dividere per 2 il numero ottenuto dalla divisione precedente, tenere il resto, … finchè non si ottiene il valore 0 come risultato della divisione. La codifica binaria è rappresentata dai resti presi in ordine inverso.

<u>Esercizio 1.1 Convertire il numero decimale 17 in un numero binario.</u>

Per rappresentare il numero 17 abbiamo bisogno di utilizzare almeno 5 bit. Infatti utilizzando una codifica con quattro bit prossiamo rappresentare al massimo il valore 2^4 (16) − 1 = 15. Utilizzando una codifica di 5 bit il massimo valore rappresentabile sarà 2^5(32) −1 = 31. Applicando l'algoritmo si ottiene:

17/2 = 8, Resto = 1
8/2 = 4, Resto = 0
4/2 = 2, Resto = 0
2/2 = 1, Resto = 0
1/2 = 0, Resto = 1

La codifica binaria si ottiene prendendo dall'ultimo al primo i resti delle divisioni. Quindi la rappresentazione binaria di 17 sarà 10001. Usando un numero maggiore di bit la rappresentazione binaria di 17 si ottiene prendendo il numero appena ottenuto ed aggiungere tanti zeri alla sinistra fino a raggiungere il numero di bit richiesto. (*Esempio*: su 8 bit, 17 è 00010001).

<u>Esercizio 1.2 Convertire il numero decimale 35 in un numero binario.</u>

Dobbiamo utilizzare almeno 6 bit (2^5(32) − 1 = 31 mentre 2^6(64) − 1 = 63)

35/2 = 17, Resto = 1
17/2 = 8, Resto = 1
8/2 = 4, Resto = 0
4/2 = 2, Resto = 0
2/2 = 1, Resto = 0
1/2 = 0, Resto = 1

Quindi la rappresentazione binaria di 35 sarà 100011

<u>Esercizio 1.3 Convertire il numero decimale 140 in un numero binario.</u>

Dobbiamo utilizzare almeno 8 bit (2^6(256) − 1 = 255)

140/2 = 70, Resto = 0
70/2 = 35, Resto = 0
35/2 = 17, Resto = 1
17/2 = 8, Resto = 1
8/2 = 4, Resto = 0
4/2 = 2, Resto = 0
2/2 = 1, Resto = 0
1/2 = 0, Resto = 1

Quindi la rappresentazione binaria di 35 sarà 10001100. Si può osservare che 140=35x4 infatti, rispetto all'esercizio precedente il numero binario è il medesimo spostato di due bit a 0 aggiuntivi.

Esercizio 1.4 Convertire il numero decimale 286 in un numero binario. (9 bit - 100011110)

Dobbiamo utilizzare almeno 9 bit (2^9(512) − 1 = 511)

286/2 = 143, Resto = 0
143/2 = 71, Resto = 1
71/2 = 35, Resto = 1
35/2 = 17, Resto = 1
17/2 = 8, Resto = 1
8/2 = 4, Resto = 0
4/2 = 2, Resto = 0
2/2 = 1, Resto = 0
1/2 = 0, Resto = 1

Esericizio 1.5 Convertire il numero decimale 527 in un numero binario. (10 bit - 100001111)

527	1
263	1
131	1
65	1
32	0
16	0
8	0
4	0
2	0
1	1
0	

Esericizio 1.6 Convertire il numero decimale 104 in un numero binario. (8 bit - 1101000)

104	0
52	0
26	0
13	1
6	0
3	1
1	1
0	

Esericizio 1.7 Convertire il numero decimale 972 in un numero binario. (10 bit - 1111001100)

972	0
486	0
243	1
121	1
60	0
30	0
15	1
7	1
3	1
1	1
0	

Esericizio 1.8 Convertire il numero decimale 67 in un numero binario. (7 bit - 1000011)

67	1
33	1
16	0
8	0
4	0
2	0
1	1
0	

2. *Da binario a decimale*

Per passare dalla codifica binaria di un numero al corrispettivo valore decimale basta moltiplicare ogni cifra binaria per la corrispondente potenza 2^n, dove n è la posizione della cifa binaria nella codifica.

Esercizio 2.1 Codifica binaria 1101

$1 * 2^0 + 0 * 2^1 + 1 * 2^2 + 1 * 2^3 = 1 + 0 + 4 + 8 = 13$

Il valore nel sistema decimale è 13.

Esercizio 2.2 Codifica binaria 1101001

$1 * 2^0 + 0 * 2^1 + 0 * 2^2 + 1 * 2^3 + 0 * 2^4 + 1 * 2^5 + 1 * 2^6 = 1 + 8 + 32 + 64 = 105$

Il valore nel sistema decimale è 105.

Esercizio 2.3 Codifica binaria 10101010

$0*2^0 +1*2^1 +0*2^2 +1*2^3 +0*2^4 +1*2^5 +0*2^6+1*2^7= 2+8+32+128 = 170$

Il valore nel sistema decimale è 170.

Esercizio 2.4 Codifica binaria 01101100011

$1*2^0+1*2^1+0*2^2+0*2^3+0*2^4+1*2^5 +1*2^6 +0*2^7 +1*2^8+1*2^9+0*2^{10}=1+2+32+64+256+512 = 867$

Il valore nel sistema decimale è 867.

Esercizio 2.5 Codifica binaria 1111001100

$0*2^0+0*2^1+1*2^2+1*2^3+0*2^4+0*2^5 +1*2^6 +1*2^7 +1*2^8+1*2^9=4+8+64+128+256+512 = 972$

Esercizio 2.6 Codifica binaria 100101001100

$0*2^0+0*2^1+1*2^2+1*2^3+0*2^4+0*2^5 +1*2^6 +0*2^7 +1*2^8+0*2^9+0*2^{10}+1*2^{11}=4+8+64+256+2048= 2380$

3. *Modulo e Segno*

La codifica in modulo e segno permette di reppresentare numeri positivi e negativi. Supponendo di aver a disposizione n bit, tramite tale codifica possiamo rappresentare numeri in un intervallo compreso tra $-2^{n-1} - 1$ e $2^{n-1} - 1$. Nella codifica in modulo e segno si utilizza il bit più significativo (quello più a sinistra) per rappresentare il segno; se esso è posto a 1 tale numero è negativo, altrimenti è positivo. Le rimanenti cifre sono utilizzate per rappresentare il modulo (il numero in valore assoluto, ovvero non considerando il segno). *(Nota La codifica del numero zero è duplice. Supponendo di avere a disposizione 8 bit, entrambe le codifiche 00000000 e 10000000 rappresenteranno lo zero (non esiste differenza tra +0 e -0)*.

Esercizio 3.1 Convertire −13 da decimale a binario con la codifica in Modulo e Segno.

Il modulo di −13 (scritto $|-13|$) è 13, la cui rappresentazione binaria utilizzando 4 bit è 1101. Per rappresentare −13 sono sufficienti 4 cifre binarie per il modulo, più un bit riservato per il segno, per un totale di 5 cifre. Pertanto, ponendo il bit più significativo a 1, la rappresentazione sarà: 11101.

Esercizio 3.2 Convertire −34 da decimale a binario con la codifica in Modulo e Segno.

$|-34| = 100010$, Soluzione 1100010

34	0
17	1
8	0
4	0
2	0
1	1
0	

Esercizio 3.3 Convertire 65 da decimale a binario con la codifica in Modulo e Segno.

$|65| = 1000001$, Soluzione 01000001

65	1
32	0
16	0
8	0
4	0
2	0
1	1
0	

Esercizio 3.4 Convertire -109 da decimale a binario con la codifica in Modulo e Segno.

$|-109| = 1101101$, Soluzione 11101101

109	1
54	0
27	1
13	1
6	0
3	1
1	1
0	

Esercizio 3.5 Convertire 11111 da binario a decimale con la codifica in Modulo e Segno.

Il primo bit posto a 1 indica che il numero è negativo, le restanti cifre codificano il modulo.

Segno=1→negativo Modulo=1111=$1*2^0+1*2^1+1*2^2+1*2^3=15$ Soluzione=−15.

Esercizio 3.6 Convertire 101010 da binario a decimale con la codifica in Modulo e Segno.

Segno=1→negativo Modulo=01010=$0*2^0+1*2^1+0*2^2+1*2^3+0*2^4=10$ Soluzione=−10.

Esercizio 3.7 Convertire 010001001 da binario a decimale con la codifica in Modulo e Segno.

Segno=0→ positivo Modulo=10001001=137 Soluzione=137.

Esercizio 3.8 Convertire 1101100101 da binario a decimale con la codifica in Modulo e Segno.

Segno=1→negativo Modulo=101100101=357 Soluzione=−357.

4. Codifica in complemento a 2

La codifica in complemento a 2 permette di rappresentare numeri positivi e negativi. Supponendo di aver a disposizione n bit, tramite tale codifica possiamo rappresentare numeri in un intervallo compreso tra -2^{n-1} e $2^{n-1}-1$. Nella codifica in complemento a 2 il bit più significativo rappresenta il segno come nella precedente codifica; se esso è posto a 1 tale numero è negativo, altrimenti è positivo. Le rimanenti cifre sono utilizzate per rappresentare il modulo (il numero in valore assoluto, ovvero non considerando il segno). Nel caso della rappresentazione dei numeri positivi, la codifica corrisponde a quella in modulo e segno; nel caso dei numeri negativi si determina il complemento di quello positivo (invertendo gli 0 e gli 1 della rappresentazione positiva) e si somma 1. *(Nota La codifica del numero zero su 8 bit nel caso del complemento a 2 è unica, e sarà 00000000. Il valore 10000000 rappresenterà invece* -2^{n-1} *= −27).*

<u>Esercizio 4.1 Convertire −13 da decimale a binario con la codifica in complemento a 2.</u>

Rappresentazine binaria di 13 = 01101, il cui complemento è 10010.

$$\begin{array}{rc} 10010 & + \\ 1 & = \\ 10011 & \end{array}$$

Ripetendo il procedimento sul risultato ottentuto, ritorniamo alla codifica binaria originale del modulo.

$$\begin{array}{rc} 01100 & + \\ 1 & = \\ 01101 & \end{array}$$

<u>Esercizio 4.2 Convertire -34 da decimale a binario con la codifica in complemento a 2 utilizzando una codifica a 8 bit.</u>

Rappresentazine binaria su 8 bit di 34 = 00100010, il cui complemento è 11011101.

$$\begin{array}{rc} 11011101 & + \\ 1 & = \\ 11011110 & \end{array}$$

<u>Esercizio 4.3 Convertire 67 da decimale a binario con la codifica in complemento a 2 utilizzando una codifica a 9 bit.</u>

La rappresentazione di numeri positivi è identica alla codifica binaria tradizionale, per cui rappresentazione binaria su 9 bit di 67 è 001000011.

<u>Esercizio 4.3 Convertire -109 da decimale a binario con la codifica in complemento a 2 utilizzando una codifica a 10 bit.</u>

Rappresentazione binaria su 10 bit di 109 = 0001101101, il cui complemento è 1110010010.

$$\begin{array}{rc} 1110010010 & + \\ 1 & = \\ 1110010011 & \end{array}$$

<u>Esercizio 4.4 Convertire -321 da decimale a binario con la codifica in complemento a 2</u>

Rappresentazione binaria di 321 = 0101000001, il cui complemento è 1010111110.

$$\begin{array}{rc} 0101000001 & + \\ 1 & = \\ 1010111111 & \end{array}$$

Esercizio 4.5 Convertire 11111 da binario con la codifica in complemento a 2 a decimale

Nella codifica in complemento a 2 il bit più significativo indica il segno, per cui 11111 sarà negativo. Applichiamo lo stesso algoritmo visto negli esempi precedenti per ottenere il modulo in codifica binaria. Prendiamo il complemento di 11111 che è 00000, per cui

$$\begin{array}{rc} 00000 & + \\ 1 & = \\ 00001 & \end{array}$$

Il valore 11111 nella codifica in complemento a 2 rappresenta il decimale −1.

Esercizio 4.6 Convertire 101010 da binario con la codifica in complemento a 2 a decimale

Prendiamo il complemento di 101010 che è 010101, per cui

$$\begin{array}{rc} 010101 & + \\ 1 & = \\ 010110 & \end{array}$$

010110=$0*2^0+1*2^1+1*2^2+0*2^3+1*2^4$=2+4+16=22

Il valore 101010 nella codifica in complemento a 2 rappresenta il decimale −22.

Esercizio 4.7 Convertire 010001001 da binario con la codifica in complemento a 2 a decimale

Il bit più significativo posto a 0 indica che il numero è positivo, per cui la codifica corrisponde a quella vista per i numeri positivi. Il decimale è 137.

Esercizio 4.8 Convertire 1001100101 da binario con la codifica in complemento a 2 a decimale

Prendiamo il complemento di 1001100101 che è 0110011010, per cui

$$\begin{array}{rc} 0110011010 & + \\ 1 & = \\ 0110011011 & \end{array}$$

Il valore 1001100101 nella codifica in complemento a 2 rappresenta il decimale −411.

5. *Operazioni aritmetiche con i numeri binari*

Come per le somme decimali secondo le seguenti regole (0+0=0, 0+1=1,1+0=1, 1+1=0 riporto 1). Le sottrazioni equivalgono al sommare il numero col rispettivo complemento 2.

Esercizio 5.1 Sommare 1011(11) e 101010(42)

$$\begin{array}{rc} 1011 & + \\ 101010 & = \\ 110101 & \end{array}$$

La somma equivale a 110101(53)

Esercizio 5.2 Sommare 1001101(77) e 10110010(178)

$$\begin{array}{rl} 1001101 & + \\ 10110010 & = \\ 11111111 & \end{array}$$

La somma equivale a 11111111(255)

Esercizio 5.3 Sottrarre 101101(45) e 1011 (11)

101101 in complemento a 2 ha 7 bit diventando 0101101. 1011 viene reso negativo con il complemento a 2 è 10101 e a 7 bit è 1110101.

$$\begin{array}{rl} 0101101 & + \\ 1110101 & = \\ 10100010 & \end{array}$$

La differenza equivale a 0100010(34)

Esercizio 5.4 Sommare 00010000(16) e 11010110 (-42)

$$\begin{array}{rl} 00010000 & + \\ 11010110 & = \\ 11100110 & \end{array}$$

La somma equivale a 11100110 (-26)

Esercizio 5.5 Sommare 111011(-5) e 100100 (-28)

$$\begin{array}{rl} 111011 & + \\ 100100 & = \\ 1011111 & \end{array}$$

La somma equivale a (1)011111 (modulo di 011111 è 00001 1 che è diverso da -33)

Esercizio 5.6 Sommare 0011011(27) e 0111101 (61)

$$\begin{array}{rl} 0011011 & + \\ 0111101 & = \\ 1011000 & \end{array}$$

La somma equivale a 1011000 (negativo diverso da 88)

6. *Codifica ottale e esadecimale*

Per convertire da decimale a ottale (o esadecimale) e viceversa valgono le stesse regole viste per la codifica binaria.

Esercizio 6.1 Convertire 51 da decimale a ottale.

51	3
6	6
0	

63(ott) = $6*8^1+3*8^0$=48+3

Esercizio 6.2 Convertire 414 da decimale a esadecimale.

414	14
25	9
1	1
0	

A=10, B=11, C=12, D=13, E=14, F=15

19E(hex) = 1*16^2+9*16^1+E*16^0=256+9*16+14 = 256+160-16+14=414

Esercizio 6.3 Convertire 147 da ottale a binario.

Per convertire da ottale (o esadecimale) a binario ciascuna cifra viene scritta in codice binario a 3(o 4) bit e poi vengono assemblate.

147 → 1: 001, 4: 100, 7: 111 → 001100111

Esercizio 6.4 Convertire 62D4 da esadecimale a binario.

62D4 → 6: 0110, 2: 0010, D:1101, 4: 0100 → 0110001011010100

Esercizio 6.5 Convertire 101110001 da binario a ottale.

Per convertire da binario a ottale (o esadecimale) le cifre vengono raggruppate a 3 (o 4) bit e tradotte singolarmente.

101110001→ 101: 5, 110:6, 001:1

Esercizio 6.6 Convertire 100111100101 da binario a esadecimale.

100111100101→ 1001:9, 1110:14:E, 0101:5→ 9E5