

Rappresentazione dei numeri e conversioni con foglio di calcolo

ESERCITAZIONE N. 1 DI LABORATORIO:

Utilizzando un foglio di calcolo, realizzare un sistema che effettui la conversione in decimale di un numero binario su 8 bit, applicando la formula generale di conversione.

Soluzione:

Per calcolare la conversioni da binario a decimale occorre applicare la seguente formula:

$$\sum_{k=0}^{n-1} C_k * B^k = C_{n-1} * B^{n-1} + C_{n-2} * B^{n-2} + \dots + C_2 * B^2 + C_1 * B^1 + C_0 * B^0$$

La formula deve essere inserita in un formato comprensibile al foglio elettronico. Per questo occorre caricare nelle celle da A4 a H4 le potenze della base 2. La riga 5 viene utilizzata per inserire i valori. Nella cella J5 viene immessa la seguente formula: =H5*H4+G5*G4+F5*F4+E5*E4+D5*D4+C5*C4+B5*B4+A5*A4

	A	B	C	D	E	F	G	H	I	J
1	Formula generale									
2			Base	2						
3										
4	128	64	32	16	8	4	2	1	=	decimale
5	1	1	1	1	1	1	1	1	=	255
6										

Variando i bit della riga 5, il sistema di conversione fornisce il corrispondente valore decimale.

Uso Calc:

	A	B	C	D	E	F	G	H	I	J
1	Conversione da binario (parola di 8 bit) in decimale									
2										
3	base	2								
4	128	64	32	16	8	4	2	1		decimale
5	1	1	1	1	1	1	1	1	corrisponde a	255
6										
7	Esempio: parola									
8										
9	0	0	0	0	1	1	1	1		15
10										
11	Uso della formula generale di conversione									
12										
13										
14										
15										
16										
17										

nb: il foglio di calcolo mette a disposizione le funzioni di conversione da una base numerica all'altra

BINARIO.DECIMALE
BINARIO.HEX
BINARIO.OCT

DECIMALE.BINARIO
DECIMALE.HEX
DECIMALE.OCT

HEX.BINARIO
HEX.DECIMALE
HEX.OCT

OCT.BINARIO
OCT.DECIMALE
OCT.HEX

ESERCITAZIONE

Dato un numero decimale, dopo aver effettuato la conversione in binario, rappresentare il numero in esadecimale, usando un foglio di calcolo.

Soluzione:

Per convertire il numero da decimale a binario si può usare il metodo delle divisioni successive che consiste nel dividere per due il numero da convertire, prelevare il resto ed allinearli nei bit da destra verso sinistra.

In Generale: dato un numero N in base 10, le cifre in una qualsiasi base B sono i resti delle divisioni successive di N per la base di arrivo B. Il resto del primo rapporto corrisponde alla cifra meno significativa

Funzioni utilizzate:

Per calcolare il resto delle divisioni è stata utilizzata la funzione **RESTO** che segue:

$$=RESTO(B10;D10)$$

Uso Calc:

21	Conversione da decimale in binario (parola di 8 bit)								
22									
23	Per divisioni successive: i resti interi sono visualizzati								
24									
25	N decimale	N/2	N/4	N/8	N/16	N/32	N/64	N/128	
26	15	7,5	3,75	1,88	0,94	0,47	0,23	0,12	
27									
28	resti troncati a interi								
29									
30	0	0	0	0	1	1	1	1	binario
31									
32	TRONCA (RESTO(H26;2))							TRONCA (RESTO(A26;2))	

Per troncare ad intero :

$$=TRONCA (RESTO(cella;2))$$

34	Conversione da binario (BCD) a ESADECIMALE								
35									
36	Per raggruppamento; se numero maggiore di nove si usano lettere dell'alfabeto da A a F								
37									
38	Ogni cifra ESADECIMALE								
39									
40									
41									
42	BCD								
43	8	4	2	1	8	4	2	1	
44	1	1	1	1	1	1	1	1	
45									corrisponde a
46									
47									cifra più
48									significativa
49									15
50									ESADECIMALE
51									F

Rappresentazione dei numeri e conversioni in C++

ESERCITAZIONE

Scrivere un programma che immesso un numero decimale da tastiera, lo rappresenti in complemento a uno.

Soluzione:

Il programma viene implementato con il linguaggio di programmazione C++. Per rappresentare un numero decimale in complemento ad uno occorre innanzitutto trovare il valore assoluto e convertirlo in binario. Per la conversione viene usato il metodo delle divisioni successive.

La rappresentazione in complemento a 1 si effettua semplicemente invertendo tutte le cifre del numero binario.

Per calcolare il resto delle divisioni è stata utilizzata la funzione `%` come segue:

```
b[i]=n%2;  
n=n/2;
```

Mentre i resti delle divisioni sono stati memorizzati nel vettore `b[i]`.

Per calcolare il complemento a uno è stata usata la funzione `if` come segue:

```
if (b[i]==0)  
b[i]=1;  
else b[i]=0;
```

Listing

```
#include <cstdlib>  
#include <iostream>  
#include <cmath>  
using namespace std;  
int main()  
{  
    int Dim = 8;  
    int n, b[Dim];  
    cout << "\nInserisci un numero intero in sistema decimale ... \n\n";  
    cin >> n;  
    if (n < 0)  
        n = abs(n); // controllo sull'inserimento (solo positivi)  
    for (int i = Dim - 1; i >= 0; i--)  
    {  
        b[i] = n % 2; // operatore resto o modulo  
        n = n / 2;  
    }  
    cout << "\nIl numero convertito in binario e': \n\n";  
    for (int i = 0; i < Dim; i++)  
        cout << b[i];  
    cout << "\n\nComplemento a uno \n\n";  
    for (int i = 0; i < Dim; i++)  
    {  
        if (b[i] == 0)  
            b[i] = 1;  
        else  
            b[i] = 0;  
        cout << b[i];  
    }  
    cout << endl << endl;  
    system("PAUSE");  
    return EXIT_SUCCESS;  
}
```

Inserisci un numero intero in sistema decimale ...

15

Il numero convertito in binario e':

00001111

Complemento a uno

11110000

Premere un tasto per continuare . . . _