

Conversioni e operazioni binarie e con altre basi

December 4, 2024

In questo notebook Jupyter Python sono presenti le principali conversioni e operazioni che si possono eseguire/calcolare con i diversi sistemi di numerazione (decimale, ottale, binario, esadecimale, ecc...)

1 LIBRERIE NECESSARIE

```
[ ]: import time
```

2 CONVERTITORE DI NUMERI DA QUALSIASI BASE ALLA BASE DECIMALE

```
[ ]: def convertitore_numeri_da_base_qualsiasi_a_base_decimale():  
    print("Benvenuto nel convertitore di numeri da qualsiasi base alla base_  
    ↳decimale")  
    numero_da_convertire=input("Inserisci il numero che desideri venga_  
    ↳convertito al decimale: ").upper()  
    base_numero_da_convertire=int(input("Inserisci la base in numero (binaria,_  
    ↳ottale, esadecimale, ecc...) del numero che desideri venga convertito al_  
    ↳decimale: "))  
    cifre_sistema_esadecimale={"A":10,"B":11,"C":12,"D":13,"E":14,"F":15,"0":  
    ↳0,"1":1,"2":2,"3":3,"4":4,"5":5,"6":6,"7":7,"8":8,"9":9,"10":10}  
    lista_cifre_numero_da_convertire=[cifre_sistema_esadecimale[cifra] for_  
    ↳cifra in numero_da_convertire]  
    ↳  
    ↳lista_invertita_cifre_numero_da_convertire=lista_cifre_numero_da_convertire[:  
    ↳:-1]  
    print(f"Il numero indicato inserito è: {numero_da_convertire}, quindi le_  
    ↳cifre del numero sono: {lista_cifre_numero_da_convertire}")  
    print(f"Per prima cosa bisogna analizzare il numero e ordinare le cifre al_  
    ↳contrario, quindi le cifre al contrario del numero inserito sono:_  
    ↳{lista_invertita_cifre_numero_da_convertire}")  
    print(f"Poi bisogna moltiplicare ogni cifra per la base_  
    ↳{base_numero_da_convertire} elevata alla posizione della cifra stessa,_  
    ↳partendo da 0 per la cifra meno significativa (quella a destra)")  
    i=-1  
    calcolo_finale_cifre=[]
```

```

    □
    ↪ lista_cifre_numero_da_convertire=[cifre_sistema_esadecimale[cifra_esadecimale]
    ↪ for cifra_esadecimale in numero_da_convertire]
        for cifra_originale in lista_invertita_cifre_numero_da_convertire:
            i+=1
            cifra_convertite=cifra_originale*(base_numero_da_convertire**i)
            calcolo_finale_cifre.append(cifra_convertite)
        numero_convertito=sum(calcolo_finale_cifre)
        print(f"Quindi, il numero {numero_da_convertire} in base
    ↪ {base_numero_da_convertire} corrisponde a {numero_convertito} in base
    ↪ decimale.")
convertitore_numeri_da_base_qualsiasi_a_base_decimale()

```

Benvenuto nel convertitore di numeri da qualsiasi base alla base decimale
 Il numero indicato inserito è: 1011010, quindi le cifre del numero sono: [1, 0, 1, 1, 0, 1, 0]
 Per prima cosa bisogna analizzare il numero e ordinare le cifre al contrario, quindi le cifre al contrario del numero inserito sono: [0, 1, 0, 1, 1, 0, 1]
 Poi bisogna moltiplicare ogni cifra per la base 2 elevata alla posizione della cifra stessa, partendo da 0 per la cifra meno significativa (quella a destra)
 Quindi, il numero 1011010 in base 2 corrisponde a 90 in base decimale.

3 CONVERTITORE DI NUMERI DALLA BASE DECIMALE AD UN'ALTRA BASE QUALSIASI

```

[ ]: def convertitore_numeri_da_base_decimale_a_base_qualsiasi():
    print("Benvenuto nel convertitore di numeri dalla base decimale ad un'altra
    ↪ base qualsiasi")
    numero_da_convertire=input("Inserisci il numero che desideri venga
    ↪ convertito dalla base decimale ad un'altra base qualsiasi: ").upper()
    base_numero_convertito=int(input("Inserisci la base in numero (binaria,
    ↪ ottale, esadecimale, ecc...) di cui desideri venga convertito il numero in
    ↪ questione: "))
    print("Per effettuare la conversione, bisogna eseguire delle divisioni
    ↪ successive per la base del numero da convertire. Il numero convertito si
    ↪ ottiene leggendo i resti al contrario")
    lista_resti_divisione_ripetuta=[]
    numero_da_convertire=int(numero_da_convertire)
    quoziente_divisione_ripetuta=numero_da_convertire
    while quoziente_divisione_ripetuta!=0:
        □
        ↪ resto_divisione_ripetuta=quoziente_divisione_ripetuta%base_numero_convertito
        if base_numero_convertito==16 and resto_divisione_ripetuta>=10:
            resto_divisione_ripetuta=chr(55 + resto_divisione_ripetuta)
        lista_resti_divisione_ripetuta.append(str(resto_divisione_ripetuta))

```

```

        quoziente_divisione_ripetuta=quoziente_divisione_ripetuta//
        ↪base_numero_convertito
        lista_invertita_resti_divisione_ripetuta=lista_resti_divisione_ripetuta[:
        ↪-1]
        numero_convertito="".join(lista_invertita_resti_divisione_ripetuta)
        print(f"Quindi, il numero {numero_da_convertire} in base decimale
        ↪corrisponde a {numero_convertito} in base {base_numero_convertito}")
convertitore_numeri_da_base_decimale_a_base_qualsiasi()

```

Benvenuto nel convertitore di numeri dalla base decimale ad un'altra base qualsiasi

Per effettuare la conversione, bisogna eseguire delle divisioni successive per la base del numero da convertire. Il numero convertito si ottiene leggendo i resti al contrario

Quindi, il numero 2 in base decimale corrisponde a 10 in base 2

4 CONVERTITORE DI NUMERI DA UNA QUALSIASI BASE NON DECIMALE AD UN'ALTRA BASE NON DECIMALE

```

[ ]: def convertitore_numeri_da_base_qualsiasi_a_base_qualsiasi():
    print("Benvenuto nel convertitore di numeri da qualsiasi base ad un numero
    ↪con qualsiasi base!")
    cifre_sistema_esadecimale={"A":10,"B":11,"C":12,"D":13,"E":14,"F":15,"0":
    ↪0,"1":1,"2":2,"3":3,"4":4,"5":5,"6":6,"7":7,"8":8,"9":9}
    numero_da_convertire=input("Inserisci il numero che desideri venga
    ↪convertito: ").upper()
    base_numero_da_convertire=int(input("Inserisci la base del numero originale
    ↪(es. 2, 8, 10, 16): "))
    base_numero_convertito=int(input("Inserisci la base del numero di cui
    ↪desideri venga convertito il numero (es. 2, 8, 10, 16): "))
    if base_numero_da_convertire==base_numero_convertito:
        print("Le due basi inserite sono uguali. Riprovare inserendo basi
        ↪diverse.")
    convertitore_numeri_da_base_qualsiasi_a_base_qualsiasi()
    numero_decimale=0
    for i,cifra in enumerate(numero_da_convertire):
        ↪
    ↪numero_decimale=numero_decimale+cifre_sistema_esadecimale[cifra]*(base_numero_da_convertire
    if numero_decimale==0:
        numero_convertito="0"
    else:
        cifre="0123456789ABCDEF"
        numero_convertito=""
        while numero_decimale>0:
            ↪
    ↪numero_convertito=cifre[numero_decimale%base_numero_convertito]+numero_convertito

```

```

        numero_decimale=numero_decimale//base_numero_convertito
    print(f"Il numero {numero_da_convertire} con base_
↪{base_numero_da_convertire} convertito in base {base_numero_convertito}_
↪corrisponde a {numero_convertito}")
convertitore_numeri_da_base_qualsiasi_a_base_qualsiasi()

```

Benvenuto nel convertitore di numeri da qualsiasi base ad un numero con qualsiasi base!

Il numero 1011010 con base 2 convertito in base 8 corrisponde a 55

5 CALCOLATORE DEL COMPLEMENTO A 1 (CA1) DI UN NUMERO BINARIO

```

[ ]: def complemento_a_1():
    print("Benvenuto nel calcolatore del CA1 (complemento a 1) di un numero_
↪binario")
    numero_originale=int(input("Inserisci il numero binario di cui desideri_
↪calcolare il CA1 (complemento a 1) del numero in questione: "))
    print("Per calcolare il CA1 (complemento a 1) di un qualsiasi numero_
↪binario basta trasformare ogni 0 in 1 e viceversa")
    lista_bit_numero_da_calcolare=[int(bit) for bit in str(numero_originale)]
    lista_bit_ribaltata_numero_da_calcolare=lista_bit_numero_da_calcolare
    for i in range(len(lista_bit_ribaltata_numero_da_calcolare)):
        if lista_bit_ribaltata_numero_da_calcolare[i]==0:
            lista_bit_ribaltata_numero_da_calcolare[i]=1
        else:
            lista_bit_ribaltata_numero_da_calcolare[i]=0
    numero_CA1=map(str, lista_bit_ribaltata_numero_da_calcolare)
    numero_CA1="".join(numero_CA1)
    print(f"Quindi, il CA1 (complemento a 1) del numero binario_
↪{numero_originale} è {numero_CA1}")
complemento_a_1()

```

Benvenuto nel calcolatore del CA1 (complemento a 1) di un numero binario

Per calcolare il CA1 (complemento a 1) di un qualsiasi numero binario basta trasformare ogni 0 in 1 e viceversa

Quindi, il CA1 (complemento a 1) del numero binario 101111101 è 010000010

6 CALCOLATORE DEL COMPLEMENTO A 2 (CA2) DI UN NUMERO BINARIO

```

[ ]: def complemento_a_2():
    print("Benvenuto nel calcolatore del CA2 (complemento a 2) di un numero_
↪binario")
    numero_originale=int(input("Inserisci il numero binario di cui desideri_
↪calcolare il CA2 del numero in questione: "))

```

```

    print("Per calcolare il CA2 (complemento a 2) di un qualsiasi numero_
↪binario bisogna prima calcolare il valore del CA1 (complemento a 1) e poi_
↪sommare a quest'ultimo 1 in binario")
    lista_bit_numero_da_calcolare=[int(bit) for bit in str(numero_originale)]
    lista_bit_ribaltata_numero_da_calcolare=lista_bit_numero_da_calcolare
    for i in range(len(lista_bit_ribaltata_numero_da_calcolare)):
        if lista_bit_ribaltata_numero_da_calcolare[i]==0:
            lista_bit_ribaltata_numero_da_calcolare[i]=1
        else:
            lista_bit_ribaltata_numero_da_calcolare[i]=0
    numero_CA1=map(str, lista_bit_ribaltata_numero_da_calcolare)
    numero_CA1="".join(numero_CA1)
    numero_CA2=int(numero_CA1, 2)
    numero_CA2=numero_CA2+1
    numero_CA2=bin(numero_CA2)
    numero_CA2=numero_CA2[2:]
    print(f"Quindi, il complemento a 2 (CA2) del numero binario_
↪{numero_originale} è {numero_CA2}")
complemento_a_2()

```

Benvenuto nel calcolatore del CA2 (complemento a 2) di un numero binario
 Per calcolare il CA2 (complemento a 2) di un qualsiasi numero binario bisogna
 prima calcolare il valore del CA1 (complemento a 1) e poi sommare a quest'ultimo
 1 in binario
 Quindi, il complemento a 2 (CA2) del numero binario 101111101 è 10000011

7 MENÙ DI SCELTA DEL PROGRAMMA DESIDERATO

```

[ ]: def menù_scelta_programma():
    print("Benvenuto, con questo tool è possibile effettuare le principali_
↪conversioni e operazioni che si possono eseguire/calcolare con i diversi_
↪sistemi di numerazione (decimale, ottale, binario, esadecimale, ecc...)")
    programma_scelto=input("""Selezionare il programma più adatto alle proprie_
↪esigenze (scrivere il numero associato):\n
    1) Convertitore di numeri da qualsiasi base alla base decimale\n
    2) Convertitore di numeri dalla base decimale ad un'altra base qualsiasi\n
    3) Convertitore di numeri da una qualsiasi base non decimale ad un'altra_
↪base non decimale\n
    4) Calcolatore del complemento a 2 (CA2) di un numero binario\n
    5) Calcolatore del complemento a 2 (CA2) di un numero binario""")
    print(f"È stato selezionato il programma numero {programma_scelto}")
    if programma_scelto=="1":
        convertitore_numeri_da_base_qualsiasi_a_base_decimale()
        riavvio_programma=input("Si desidera selezionare nuovamente un_
↪programma di questo tool? (rispondere solo con s/n)")
        if riavvio_programma=="s":

```

```

        print("Il tool verrà riavviato a momenti")
        menù_scelta_programma()
        time.sleep(1)
    elif riavvio_programma=="n":
        print("Grazie per aver usato questo tool, il programma verrà chiuso_
↪a momenti. Alla prossima!")
        time.sleep(3)
    else:
        print("Risposta non riconosciuta! Il tool verrà riavviato comunque")
        menù_scelta_programma()
        time.sleep(1)
    elif programma_scelto=="2":
        convertitore_numeri_da_base_decimale_a_base_qualsiasi()
        riavvio_programma=input("Si desidera selezionare nuovamente un_
↪programma di questo tool? (rispondere solo con s/n)")
        if riavvio_programma=="s":
            print("Il tool verrà riavviato a momenti")
            menù_scelta_programma()
            time.sleep(1)
        elif riavvio_programma=="n":
            print("Grazie per aver usato questo tool, il programma verrà chiuso_
↪a momenti. Alla prossima!")
            time.sleep(3)
        else:
            print("Risposta non riconosciuta! Il tool verrà riavviato comunque")
            menù_scelta_programma()
            time.sleep(1)
    elif programma_scelto=="3":
        convertitore_numeri_da_base_qualsiasi_a_base_qualsiasi()
        riavvio_programma=input("Si desidera selezionare nuovamente un_
↪programma di questo tool? (rispondere solo con s/n)")
        if riavvio_programma=="s":
            print("Il tool verrà riavviato a momenti")
            menù_scelta_programma()
            time.sleep(1)
        elif riavvio_programma=="n":
            print("Grazie per aver usato questo tool, il programma verrà chiuso_
↪a momenti. Alla prossima!")
            time.sleep(3)
        else:
            print("Risposta non riconosciuta! Il tool verrà riavviato comunque")
            menù_scelta_programma()
            time.sleep(1)
    elif programma_scelto=="4":
        complemento_a_1()
        riavvio_programma=input("Si desidera selezionare nuovamente un_
↪programma di questo tool? (rispondere solo con s/n)")

```

```

    if riavvio_programma=="s":
        print("Il tool verrà riavviato a momenti")
        menù_scelta_programma()
        time.sleep(1)
    elif riavvio_programma=="n":
        print("Grazie per aver usato questo tool, il programma verrà chiuso
↳a momenti. Alla prossima!")
        time.sleep(3)
    else:
        print("Risposta non riconosciuta! Il tool verrà riavviato comunque")
        menù_scelta_programma()
        time.sleep(1)
    elif programma_scelto=="5":
        complemento_a_2()
        riavvio_programma=input("Si desidera selezionare nuovamente un
↳programma di questo tool? (rispondere solo con s/n)")
        if riavvio_programma=="s":
            print("Il tool verrà riavviato a momenti")
            menù_scelta_programma()
            time.sleep(1)
        elif riavvio_programma=="n":
            print("Grazie per aver usato questo tool, il programma verrà chiuso
↳a momenti. Alla prossima!")
            time.sleep(3)
        else:
            print("Risposta non riconosciuta! Il tool verrà riavviato comunque")
            menù_scelta_programma()
            time.sleep(1)
    else:
        print("Il programma selezionato non è esistente. Riprovare con un
↳programma presente nell'elenco iniziale")
        menù_scelta_programma()
        time.sleep(1)
menù_scelta_programma()

```

Benvenuto, con questo tool è possibile effettuare le principali conversioni e operazioni che si possono eseguire/calcolare con i diversi sistemi di numerazione (decimale, ottale, binario, esadecimale, ecc...)

È stato selezionato il programma numero 5

Benvenuto nel calcolatore del CA2 (complemento a 2) di un numero binario

Per calcolare il CA2 (complemento a 2) di un qualsiasi numero binario bisogna prima calcolare il valore del CA1 (complemento a 1) e poi sommare a quest'ultimo 1 in binario

Quindi, il complemento a 2 (CA2) del numero binario 111101 è 11

Grazie per aver usato questo tool, il programma verrà chiuso a momenti. Alla prossima!