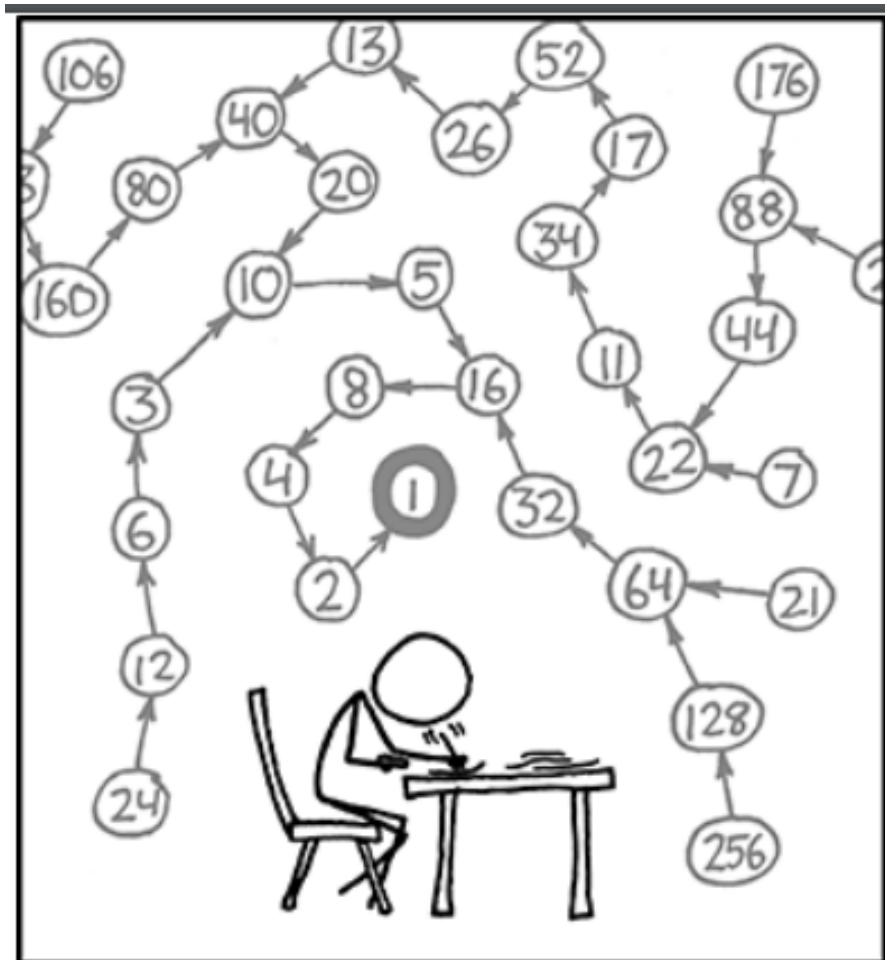


Concetti di programmazione

(con il linguaggio Python - parte 4)

Caso di studio: la congettura di Collatz

L'esercizio proposto consiste nell'implementare in Python un algoritmo molto semplice dal risultato finale sorprendente!



THE COLLATZ CONJECTURE STATES THAT IF YOU PICK A NUMBER, AND IF IT'S EVEN DIVIDE IT BY TWO AND IF IT'S ODD MULTIPLY IT BY THREE AND ADD ONE, AND YOU REPEAT THIS PROCEDURE LONG ENOUGH, EVENTUALLY YOUR FRIENDS WILL STOP CALLING TO SEE IF YOU WANT TO HANG OUT.

L'algoritmo in sintesi è il seguente:

1. **Sia dato in input un intero positivo n qualsiasi;**
2. **se n è pari lo si divida per 2;**
3. **altrimenti se n è dispari, lo si moltiplichi per 3 e si aggiunga 1.**
4. **Si produca una sequenza di interi positivi (successione) ripetendo i passaggi 2 e 3.**

Per esempio con l'input iniziale 6, pari, otterremo in output $6/2 = 3$, che essendo dispari, produrrà il valore successivo $3*3 + 1 = 10$, a cui riapplicheremo la regola per i pari ottenendo $10/2 = 5$. La successione continuerà con i valori 16, 8, 4, 2, 1. Se non si pone la condizione di terminazione $n=1$, la successione percorrerà un ciclo infinito: 1, 4, 2, 1, 4,...

Posta invece la suddetta condizione di terminazione sequenza finita di numeri sarà : 6, 3, 10, 5, 16, 8, 4, 2, 1.

In notazione matematica la funzione che produce la successione è questa:

$$f(n) = \begin{cases} n/2 & \text{if } n \equiv 0 \pmod{2} \\ 3n + 1 & \text{if } n \equiv 1 \pmod{2}. \end{cases}$$

Figura 1

Verifica sperimentale della congettura

Una congettura è un'affermazione o un giudizio fondato sull'intuito, ritenuto probabilmente vero, ma non ancora rigorosamente dimostrato, cioè dunque relegato solamente a rango di ipotesi. (Wikipedia)

Nel seguito è proposta una semplice applicazione in Python che implementa la verifica sperimentale della congettura di Collatz (l'applicazione in realtà implementa e confronta tre diverse soluzioni del problema).

Gli obiettivi didattici perseguiti sono i seguenti:

- A. **affrontare la questione, spesso non ben compresa, della progettazione del codice secondo il paradigma procedurale:**
 1. **individuazione di sottoproblemi e relative funzioni risolutive;**

2. definizione dei compiti da delegare alla parte main del codice e alle funzioni richiamate in essa;
 3. definizione delle interfacce mondo “esterno-programma” (input/output), “main-funzioni” (passaggio parametri/return).
- B. Proporre diverse soluzioni algoritmiche o di codifica dello stesso problema.

```
def collatzTest(k):
    if k % 2 == 0:
        k //= 2
    else:
        k = 3*k + 1
    return k

def collatz1(k):
    s = str(k)
    c = 0
    while k > 1:
        if k % 2 == 0:
            k //= 2
        else:
            k = 3*k + 1
        s = s + " " + str(k)
        c += 1
    return "Successione:\n" + s + "\nNumero termini: " + str(c)

def collatz2(m):
    s = str(m)
    c = 0
    while m > 1:
        m = collatzTest(m)
        s = s + " " + str(m)
        c += 1
    return "Successione:\n" + s + "\nNumero termini: " + str(c)

"""
----- MAIN -----
La parte MAIN del codice gestisce l'interazione con l'utente (input/print) e
realizza le diverse implementazioni dell'algoritmo tramite le opportune chiamate
delle funzioni precedentemente definite.
"""

n = int(input("Input per testare la congettura di Collatz: ")) # input dall'utente
# Soluzione 1: usa collatz1() per gestire iterazione e calcolo dei termini in sequenza
print("\nSoluzione 1")
print(collatz1(n)) # output verso l'utente

# Soluzione 2: usa collatz2() che a sua volta chiama collatzTest() nella sua definizione
print("\nSoluzione 2")
print(collatz2(n)) # output verso l'utente

# Terza soluzione: iterazione e calcolo dei termini in main con l'uso di collatzTest()
print("\nSoluzione 3")
succ = str(n)
cont = 0
while n > 1:
    n = collatzTest(n)
    succ = succ + " " + str(n)
    cont += 1
print("Successione:\n" + succ + "\nNumero termini:", cont) # output verso l'utente
```

Figura 2

Il paradigma procedurale

La programmazione procedurale è un paradigma di programmazione che consiste nel creare dei blocchi di codice sorgente, identificati da un nome e racchiusi da dei delimitatori, che variano a seconda del linguaggio di programmazione; questi sono detti anche sottoprogrammi (in inglese subroutine), procedure o funzioni, a seconda del linguaggio e dei loro ruoli all'interno del linguaggio stesso.

Questi blocchi possono essere dotati di parametri, cioè variabili locali i cui valori vengono forniti o passati dall'esterno del blocco di codice ed eventualmente esportati; ... All'interno di un sottoprogramma possono essere generalmente definite o dichiarate delle variabili locali, che vengono deallocate al termine del sottoprogramma stesso; il loro contenuto viene quindi perso se non salvato o trasmesso altrimenti.

In Pascal, le procedure sono distinte dalle funzioni perché non restituiscono un valore associato al nome della procedura stessa, e non possono quindi apparire a destra di una istruzione di assegnazione; i blocchi di codice sono racchiusi tra un Begin ed un End; (parole riservate, il punto e virgola fa parte dell'istruzione). (Wikipedia).

Le soluzioni proposte

In figura 2 vengono proposte tre soluzioni alternative valide al problema della verifica sperimentale della congettura di Collatz. Tali soluzioni adottano tre diverse assegnazioni dei compiti a porzioni del codice e con la conseguente definizioni di funzioni e loro utilizzo nella parte main del programma.

I compiti individuati sono:

1. input/output da e verso il “mondo esterno”, compito in genere assegnato alla parte main del programma tramite le funzioni Python input() e print();
2. implementazione della funzione in figura 1, detta funzione di Collatz o funzione C;
3. iterazione dell'applicazione della funzione C con la conseguente produzione della sequenza dei valori, dal numero in input fino alla valore terminale (1).

Esercizi

Esercizio 1

Utilizzando a piacere una delle soluzioni proposte per la verifica della congettura di Collatz, creare un programma che consenta all'utente di sottoporre una sequenza di valori in input terminata con l'invio (input di stringa vuota)

Esercizio 2

La congettura di Collatz si basa funzione C (figura 1) in cui , nel caso di argomento dispari si applica il calcolo

$$f(n) = p * n + 1 \text{ con } p = 3$$

Ora se si volesse verificare il comportamento della successione adottando una funzione in cui p è un parametro da specificare in input come dovrebbe essere modificato il programma?