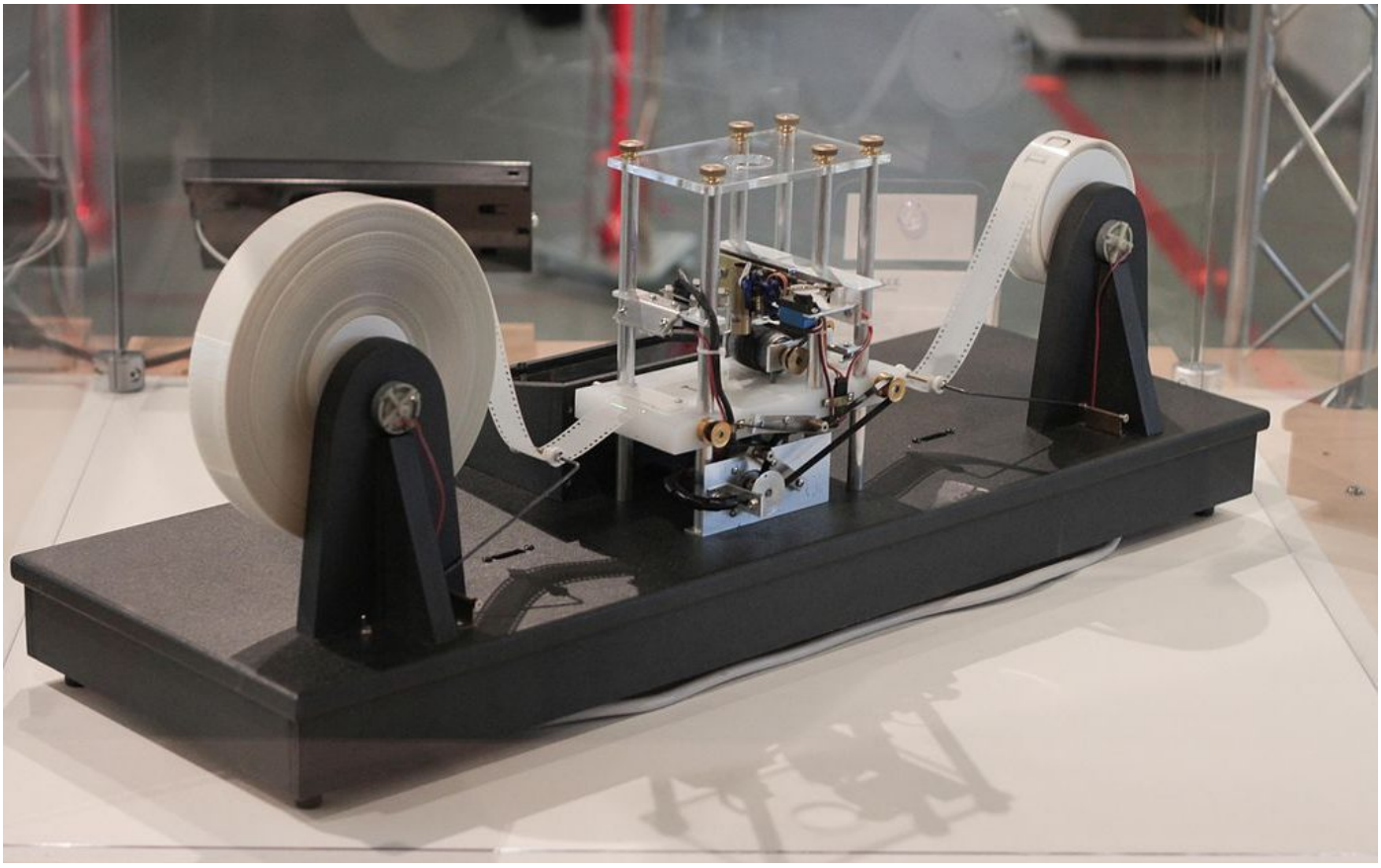

Le frontiere della computazione I | I limiti delle macchine di Turing

Cara lettrice, caro lettore,
i computer che utilizziamo quotidianamente sono basati su dei modelli classici, per esempio le macchine di Turing delle quali abbiamo parlato in questo articolo di Rivista. Le macchine di Turing sono semplici a sufficienza da poter essere realizzate nei moderni computer. Inoltre offrono una potenza di calcolo notevole, che ci permette di realizzare modelli di fenomeni complessi come il meteo, l'andamento dei mercati o la gestione dei razzi spaziali. Le macchine di Turing, però, hanno dei limiti che non possono essere superati nemmeno con le più avanzate innovazioni tecnologiche. Per esempio, alcuni problemi non si possono risolvere con le macchine di Turing. Altri, invece, si possono risolvere ma con algoritmi molto lenti, che oggi non sappiamo ancora se si possano ottimizzare. Pensiamo a due esempi: l'addizione di due numeri naturali e la risoluzione di un Sudoku.

PROBLEMI FACILI...

Per il primo problema conosciamo una procedura risolutiva semplice: la somma in colonna. Se sommiamo due numeri di una cifra compiamo un'operazione. Se sommiamo due numeri da dieci cifre ciascuno, compiamo al massimo venti

operazioni (tenendo conto dei riporti). Se i due numeri avessero diecimila cifre, compiremmo al massimo ventimila operazioni. Per descrivere questa situazione possiamo utilizzare una funzione che, sulla base della lunghezza dei due numeri in input, ci dice il numero massimo di operazioni che dobbiamo compiere per eseguire la somma. In questo caso, $f(n) \approx 2n$. I problemi la cui risoluzione richiede un algoritmo che compie un numero di operazioni polinomiale nella lunghezza dell'input n sono considerati semplici e i loro tempi di risoluzione sono brevi.

...E PROBLEMI DIFFICILI

Ora passiamo al secondo problema, quello del Sudoku. Il Sudoku è un gioco in cui è presente una griglia di 9×9 celle ognuna delle quali può contenere le cifre da 1 a 9. La griglia è suddivisa in 9 righe, 9 colonne e 9 quadrati 3×3 [si può mettere qui vicino la figura allegata? è uno screenshot del libro. Se serve il pdf chiedetemelo]. In alcune celle è inserita una cifra, altre sono vuote. Lo scopo del gioco consiste nel completare la griglia inserendo i numeri mancanti, rispettando il vincolo che in ogni riga, ogni colonna e ogni quadrato 3×3 siano presenti tutte le cifre da 1 a 9, senza ripetizioni.

L'algoritmo più efficiente per risolvere questo problema consiste nell'aggiungere una cifra alla volta e verificare che i vincoli siano rispettati. Se la cifra inserita rispetta i vincoli possiamo procedere con un nuovo tentativo. Se, invece, qualche vincolo non è rispettato, torniamo indietro e riproviamo con un'altra cifra. Questo procedimento continua finché il Sudoku non è risolto.

All'apparenza questo algoritmo non sembra difficile, ma vediamo quanti modi possibili ci sono per completare un Sudoku. Ogni casella può contenere una cifra da 1 a 9, inoltre il Sudoku ha 9×9 = 81 caselle. Di conseguenza, ci sono 9^{81} possibili combinazioni. Da questo numero dovremo togliere tutte le combinazioni non valide, cioè quelle che ripetono le stesse cifre negli stessi quadrati 3×3, righe o colonne. Il numero di possibili soluzioni è quindi 6 670 903 752 021 072 936 960. Al crescere del numero n di celle di un lato della griglia, il numero di tentativi da effettuare cresce un po' più lentamente di n^{2n}. Questa funzione cresce molto più velocemente di qualsiasi polinomio, quindi il problema del Sudoku è considerato difficile e il suo tempo di risoluzione è molto lungo. Inoltre, oggi non sappiamo ancora se ci sia un algoritmo risolutivo più veloce di questo, cioè se sia possibile ottimizzare il problema del Sudoku.

I PROBLEMI DIFFICILI NEL MONDO REALE

Il Sudoku è un passatempo più o meno divertente, ma con poche applicazioni alla vita di tutti i giorni. Però ci sono alcuni problemi difficili la cui risoluzione in tempi rapidi rivoluzionerebbe, in meglio o in peggio, la nostra quotidianità. Un esempio è il problema del folding delle proteine, cioè di determinare a partire da una catena di amminoacidi la struttura tridimensionale della proteina che essi codificano. Un'applicazione concreta di questo problema è la ricerca di una cura per alcune malattie, tra cui il cancro.

Altri esempi di problemi oggi considerati difficili riguardano la sicurezza informatica. Per esempio, indovinare una password di n lettere richiede 25^n tentativi, mentre indovinare una password composta da lettere e cifre ne richiede 35^n. Anche la decifrazione dei sistemi di crittografia moderni di cui abbiamo parlato nell'articolo è un problema difficile. E questo è un bene, infatti è proprio sulla difficoltà nel risolvere questi problemi che si basano gli odierni protocolli di sicurezza informatica.

LE FRONTIERE DELLA COMPUTAZIONE

Trovare dei metodi alternativi alle macchine di Turing che permettano di risolvere dei problemi oggi considerati difficili, quindi, avrebbe tantissime ricadute pratiche. Nei prossimi tre articoli vedremo altrettante tecniche che promettono di velocizzare drasticamente la risoluzione di alcuni di questi problemi difficili.

PER APPROFONDIRE

- Una carrellata di problemi facili e difficili è presentata nelle prime 21 slide della professoressa Linda Pagli dell'Università di Pisa;
- La classificazione dei problemi e i limiti delle macchine di Turing sono discussi nel Capitolo 1 del volume del V anno di #NetGeneration.