

## Il Pensiero Computazionale

Il pensiero computazionale si basa sul potere e sui limiti dei processi di elaborazione, siano essi eseguiti da un essere umano o da un macchina. I metodi ed i modelli computazionali ci danno il coraggio di risolvere i problemi e sistemi di progettazione che nessuno di noi sarebbe stato in grado di affrontare da solo. Pensare in maniera computazionale significa fare i conti con l'enigma dell'intelligenza artificiale: Cosa possono fare gli esseri umani meglio dei computer? E cosa possono fare i computer meglio degli esseri umani? Fondamentalmente la domanda che ci poniamo è: **Che cosa è calcolabile?** Oggi, a tali domande, sappiamo rispondere solo in parte.

Il pensiero computazionale rappresenta una competenza fondamentale per tutti, non solo per gli informatici. Come per la lettura, la scrittura e l'aritmetica, dovremmo aggiungere il pensiero computazionale pensando alla capacità di analisi di ogni bambino. Mentre la stampa ha facilitato la diffusione delle "Tre R" (così li chiamano gli americani, per noi sono la lettura, la scrittura e l'aritmetica, N.d.A.), ciò che mi sento di smentire, a questo proposito, è che l'informatica ed il computer possano facilitare la diffusione del pensiero computazionale.

Il pensiero computazionale consiste nel risolvere i problemi, progettare sistemi e comprendere il comportamento umano, attingendo dai principi fondamentali dell'informatica. Il pensiero computazionale comprende anche una serie di strumenti mentali che riflettono la vastità del mondo dell'informatica.

Dovendo risolvere un particolare problema, potremmo chiederci: Quanto è difficile da risolvere? E qual è il modo migliore per risolverlo? L'informatica (gli americani usano spesso il termine "Computer Science", per indicare "Ingegneria Informatica"; qui il termine viene sempre tradotto con "Informatica", N.d.A.) in modo simile a quello che poggia su solide basi teoriche per rispondere a queste domande con precisione. Pur comprendendo le difficoltà di elaborazione legate alla potenza della macchina – cioè il dispositivo che eseguirà la soluzione. Dobbiamo considerare il set di istruzioni della macchina, i limiti delle risorse disponibili e l'ambiente operativo. **Nel risolvere un problema in modo efficace,**

potremmo ulteriormente chiederci se una soluzione approssimata è buona abbastanza, se possiamo usare i “metodi casuali” a nostro vantaggio e se falsi positivi o falsi negativi siano ammessi (un falso positivo indica un falso allarme. Un esempio, in informatica, è un antivirus che considera erroneamente dannoso un programma innocuo, un falso negativo indica la “scelta a torto” di negativo”. Un esempio in informatica è un filtro antispam che lasci erroneamente passare una email indesiderata, N.d.A.).

Il pensiero computazionale consente di riformulare un problema apparentemente difficile in uno che sappiamo come risolvere, ricorrendo anche ad una sua riduzione, inclusione, trasformazione o simulazione.

Pensiero computazionale significa pensare in modo ricorsivo. E’ un’elaborazione parallela. Si interpreta il codice come dati e i dati come il codice. Si tratta di un controllo del dato simile alla generalizzazione di una analisi dimensionale. Si tratta di riconoscere sia le virtù ed i pericoli dell’implementazione dei linguaggi di programmazione (aliasing analysis , in inglese, N.d.A.), quindi ponendo attenzione nel dare a qualcuno o a qualcosa di più che un nome.

Inoltre, dobbiamo tenere conto sia del costo e della potenza di indirizzamento indiretto e della richiesta di sistema del cliente (in gergo, la “chiamata di procedura”, N.d.A.). Si tratta di giudicare un programma non solo per la correttezza ed l’efficienza ma anche per l’estetica e il design di un sistema che viene ottimizzato per fornirgli semplicità ed eleganza.

Il pensiero computazionale impiega l’astrazione e la decomposizione quando deve approcciare ad una funzione complessa e pesante o per progettare un sistema grande e complicato. In questo modo c’è una separazione delle competenze. Si tratta scegliere una rappresentazione adeguata di un problema o di modellare gli aspetti rilevanti di un problema al fine di renderli trattabili.

In genere, si usano le condizioni invarianti (o “pre-condizioni”, N.d.A.) per descrivere il comportamento di un sistema in modo sintetico e dichiarativo. Si tratta di avere la fiducia nel potere tranquillamente utilizzare, modificare e influenzare un sistema

grande e complesso senza doverne comprendere ogni dettaglio. Occorre modularizzare il sistema in previsione di più utenti o fare “prefetching” (tecnica usata nei microprocessori per accelerare l'esecuzione dei programmi riducendo gli stati di attesa, N.d.A.) e “caching” (memorizzazione automatica di pagine web, N.d.A.) in previsione di un utilizzo futuro.

Pensiero computazionale significa anche pensare in termini di prevenzione, di protezione e di recupero prevedendo i casi peggiori, attraverso la ridondanza, il contenimento dei danni e la correzione degli errori. In gergo tecnico si chiamano “ingorgo” (in inglese, gridlock) e “stallo” (in inglese, deadlock) e “contratti di interfacce”. Quindi bisogna imparare ad evitare la “concorrenza dei processi” (cioè l'esecuzione nello stesso istante chiamato sistema a concorrenza o sistema concorrente, N.d.A.) durante la fase di sincronizzazione del sistema.

Il pensiero computazionale utilizza il pensiero euristico per trovare una soluzione. In presenza di incertezza significa pianificare, apprendere e programmare. Ma può essere ricerca, ricerca e ancora ricerca quando si ottiene come risultato un elenco di pagine web o una strategia per vincere ad un gioco o per formulare un controesempio.

Il pensiero computazionale è tipico usare una massiccia quantità di dati per incrementare le performance dell'elaborazione. In questo ambito si cercano dei compromessi tra il tempo impiegato e lo spazio a disposizione e tra la potenza di elaborazione e la capacità di memoria.

Provate a considerare questi esempi quotidiani: quando vostra figlia va a scuola la mattina lei mette nel suo zaino le cose di cui ha bisogno per la giornata; questo è quello che possiamo definire come “ottimizzazione” (prefetching) e “memorizzazione” (caching). Quando vostro figlio perdi i suoi guanti, voi gli suggerite di rifare il percorso all'indietro: questo è il “monitoraggio a ritroso”(backtracking).

A che punto si smette di noleggiare gli sci per comprarsene un paio? A questa domanda potrebbe rispondere un algoritmo online. Quale coda fare al supermercato? Questo è un problema di modellazione per sistemi multi-server. Perché il telefono

continuerà a funzionare durante un'interruzione di corrente? Si tratta di un problema d'indipendenza, affidabilità e ridondanza nel design del sistema.

Come fare una **Test di Turing** (o CAPTCHA) completamente automatizzato per riconoscere un essere umano? Allora, occorre comprendere le difficoltà nel risolvere seri problemi di Intelligenza Artificiale per sventare gli “agenti di calcolo” (gli attacchi informatici o le procedure di frode, N.d.A.).

Il pensiero computazionale sarà radicato nella vita di tutti quando parole come “algoritmo” e “precondizione” faranno parte del vocabolario di ognuno di noi; quando “non-determinismo” e “pulizia automatica della memoria” assumeranno i significati utilizzati dagli scienziati informatici; e quando gli alberi saranno disegnati a testa in giù. A questo punto dobbiamo rilevare l'influenza del pensiero computazionale sulle altre discipline. Per esempio, il “machine learning” ha trasformato la statistica.

**L'apprendimento statistico** viene utilizzato per problemi su scala, in termini di quantità di dati e di dimensione, inimmaginabile solo pochi anni fa. Gli uffici statistici di ogni tipo di organizzazione stanno assumendo sempre più scienziati informatici.

Le scuole di informatica stanno includendo o creando dei nuovi dipartimenti di statistica. Il recente interesse degli scienziati informatici alla biologia mostra che anche i biologi sono convinti di poter trarre beneficio dal pensiero computazionale.

Il contributo della cosiddetta “Computer Science” alla biologia va oltre la capacità di analizzare grandi quantità di dati alla ricerca di modelli. La speranza è che le strutture di dati e algoritmi - ossia che le nostre astrazioni e i nostri metodi computazionali - possano rappresentare la struttura delle proteine in modi che ce ne facciano comprendere la loro funzione.

La biologia computazionale sta cambiando il modo in cui i biologi pensano. Allo stesso modo, la teoria computazionale dei giochi sta cambiando il modo di pensare degli economisti; come il “nanocomputing” influenza il modo in cui i chimici pensano, e la “computazione quantistica”, il modo in cui i fisici pensano.

Questo tipo di pensiero sarà parte del set di abilità non solo per altri scienziati ma anche per tutti gli altri. La “computazione

ubiqua” rappresenta oggi quello che il pensiero computazionale rappresenterà domani. La “**computazione ubiqua**” era il sogno di ieri che oggi è diventato realtà; il pensiero computazionale è la realtà di domani.

**Che cosa è e cosa non è.**

L'informatica è lo studio del “computare” – ossia su ciò che può essere calcolato e sul come calcolarlo. Il pensiero computazionale ha quindi la seguenti caratteristiche:

a) *Tende a concettualizzare, non a programmare.*

“Scienza del computer” non significa programmare un computer. Ragionare come uno scienziato informatico significa di più che essere grado di programmare un computer. Si richiede una mente capace di pensare a più livelli di astrazione: è fondamentale, piuttosto che imparare tutto a memoria. Un'abilità fondamentale è qualcosa che ogni essere umano deve sapere per operare nella società moderna. Imparare a memoria è invece un'abilità meccanica, ripetitiva. Ironicamente, non prima che la computer science risolve la “Grande Sfida” dell'Intelligenza Artificiale, facendo pensare i computer come gli esseri umani che imparano solo a memoria.

b) *Un modo in cui gli esseri umani, non i computer, pensano.*

Il pensiero computazionale è un modo con cui gli esseri umani possono risolvere un problema; non si sta cercando di costringere l'uomo di pensare come i computer. I computer sono noiosi e monotoni; gli esseri umani sono intelligenti e fantasiosi. Noi esseri umani sappiamo creare computer che sappiano emozionare. Dotati di questi dispositivi di calcolo, noi usiamo la nostra intelligenza per affrontare i problemi che non avremmo osato mai di affrontare prima dell'era del computer, costruendo allo stesso tempo sistemi con funzionalità limitate solo dalla nostra immaginazione.

c) *Integra e combina il pensiero matematico e ingegneristico.*

L'informatica intrinsecamente attinge al pensiero matematico, dato che, come tutte le scienze, le sue basi formali poggiano sulla matematica. L'informatica attinge intrinsecamente dal pensiero ingegneristico, dato che costruiamo sistemi che interagiscono con il mondo reale. I vincoli impliciti dell'hardware forzano gli scienziati

a pensare computazionalmente, non solo matematicamente. Lasciadoli liberi di costruire mondi virtuali che ci permettono di progettare sistemi che vanno oltre il dispositivo fisico.

d) *Idee, non artefatti.*

Non saranno solamente il software e gli artefatti hardware che produciamo che saranno fisicamente presenti ovunque e toccano la nostra vita tutti i giorni, saranno i concetti computazionali quelli che, invece, useremo per affrontare e risolvere i problemi, per gestire la nostra vita quotidiana, e per comunicare e interagire con altre persone.

e) *Per tutti, in tutto il mondo.*

Il pensiero computazionale sarà realtà quando sarà così integrato negli sforzi umani tanto da scomparire come una filosofia esplicita.

**Molte persone equiparano l'informatica con la programmazione del computer.** Alcuni genitori vedono solo una stretta gamma di opportunità di lavoro per i loro figli che si laureeranno in informatica. In tanti pensano che la ricerca di base in informatica sia compiuta e che bisogna concentrarsi solo sulla parte ingegneristica. Il pensiero computazionale rappresenta una grande visione per guidare, nell'ambito della computer science, gli educatori scientifici, i ricercatori ed i professionisti come noi per cambiare l'immagine che la società ha di questo settore. Noi, in particolare, abbiamo la necessità di raggiungere il mondo delle scuole superiori, tra cui insegnanti, genitori e studenti, inviandoli due messaggi principali:

*I problemi scientifici intellettualmente stimolanti e coinvolgenti hanno bisogno di essere capiti e risolti.*

L'area di un problema e il dominio delle sue soluzioni sono limitate solo dalla nostra curiosità e creatività.

*Chi si laurea in Informatica può fare qualsiasi cosa.*

Così come chi si laurea in Lettere o Matematica può percorrere una moltitudine di differenti carriere. Idem per la computer science. Chi si laurea in informatica può proseguire una carriera in medicina, diritto, economia, politica, qualsiasi tipo di altro ambito scientifico o di ingegneria e anche nell'ambito dell'arte.

I professori di informatica dovrebbero insegnare un corso per matricole denominato "**I modi di pensare di uno scienziato informatico**", rendendolo disponibile per tutti, non solo per i laureandi in informatica.

Si dovrebbero coinvolgere anche gli studenti delle superiori allo studio dei metodi e dei modelli computazionali. Piuttosto che lamentarsi del calo di interesse per l'informatica o per il declino dei finanziamenti per la ricerca in informatica, dovremmo cercare di ispirare l'interesse del pubblico ad avventurarsi intellettualmente in questo campo. Perciò dobbiamo diffondere la gioia, i timori e la potenza dell'informatica, con l'obiettivo per rendere popolare il pensiero computazionale.