


# Capitolo IV

# Informatica e cittadino digitale: dal *coding* al *computational thinking*

## Sommario

---

**I**n questo documento rifletto sull'importanza di un'adeguata istruzione informatica nella scuola allo scopo di preparare i futuri cittadini alle sfide che la contemporanea società digitale ci pone. Dopo aver inquadrato il tema dell'educazione informatica da un punto di vista storico, discuterò la sua novità da un punto di vista scientifico – fornendo alcuni esempi concreti delle nuove prospettive che l'informatica offre per la comprensione di fenomeni naturali ed artificiali. Affronterò anche alcuni aspetti “tecnici” di ciò che caratterizza l'informatica (rimanendo ad un livello accessibile alla comprensione generale) e analizzerò  brevemente il suo impatto sociale.

## 1 Inquadramento

---

**I**nizio con una citazione che potrebbe suonare un po' come una provocazione, soprattutto in Italia, società con una tradizione culturale di stampo fortemente umanistico. È una riflessione di George Forsythe, professore universitario di Analisi Numerica, uno dei padri della formazione universitaria in informatica, in quanto fondatore del dipartimento di informatica dell'Università

\* Professore ordinario nell'Università “Tor Vergata” di Roma.

di Stanford negli USA, tra i migliori al mondo: «*Le acquisizioni più valide nell'educazione scientifica e tecnologica sono queglii strumenti mentali di tipo generale che rimangono utili per tutta la vita. Ritengo che il linguaggio naturale e la matematica siano i due strumenti più importanti in questo senso, e l'informatica sia il terzo. L'apprendimento accoppiato di matematica ed informatica ha vantaggi pedagogici, poiché i concetti di base di ciascuna materia rinforzano l'apprendimento dell'altra.*» (Forsythe 1968, p.456-457).

È passato mezzo secolo da questa valutazione ed il mondo della scienza e della tecnologia (e non solo quello) è stato profondamente cambiato dal dilagare delle applicazioni dell'informatica. Siamo ormai in una società digitale, l'evoluzione della società industriale che dal 1700 in avanti ha fortemente cambiato il nostro modo vita. Si potrebbe dibattere a lungo su quali siano davvero gli “*strumenti mentali di tipo generale utili per tutta la vita*”. Personalmente sono d'accordo con Forsythe, ma essendo un informatico e non un pedagogista lascio la discussione su questo punto agli esperti del settore.

L'informatica è per certi versi una disciplina scientifica recente, ma per altri ha diversi millenni di storia. Ricordiamo infatti che, parallelamente alla nascita della matematica l'umanità ha prodotto “dispositivi meccanici” che l'aiutassero nell'effettuare i calcoli matematici. Del primo di questi, l'abaco, se ne trovano tracce sin dal secondo millennio a.C. ed effettivamente fino al 1900 la storia dell'informatica è la storia dei dispositivi per effettuare automaticamente calcoli matematici (Ceruzzi 2013).

Solo nel secolo scorso ci si rende conto che l'automazione del calcolo può essere applicata anche al calcolo simbolico, cioè che essa fornisce capacità di elaborazione simbolica del tutto slegate dalla realtà numerica. Per quanto riguarda queste capacità di elaborazione simbolica l'informatica si è inserita nel solco di un filone della matematica che si era occupato della dimostrazione automatica dei teoremi, e che aveva portato al fondamentale risultato di Goedel sull'impossibilità per qualunque sistema assiomatico sufficientemente potente di dimostrare tutte le sue proposizioni vere. Tale risultato viene informalmente sintetizzato nel motto “la verità è più forte della dimostrabilità”, cioè esistono proposizioni vere all'interno del sistema di cui non si può dimostrare che sono vere (“dimostrare” in senso matematico, cioè derivarle logicamente da un insieme primigenio di assiomi).

Si capisce inoltre che le strabilianti capacità di elaborazione numerica fornite dalla tecnologia informatica costituiscono un nuovo meccanismo di investigazione dei fenomeni naturali, un terzo modo di

estendere la conoscenza oltre alla teoria e alla sperimentazione. Il premio nobel per la Fisica Ken Wilson è stato uno dei primi a sostenere questo punto di vista delle “scienze computazionali”. E più recentemente Denning e Rosenbloom (2009) hanno teorizzato che l’informatica costituisce il “quarto dominio della conoscenza scientifica”.

Premetto che esistono altri modi, non scientifici, di acquisire conoscenza sul mondo, ed il fatto che non siano scientifici non inficia la loro utilità. Pensate ad esempio alle varie manifestazioni delle discipline artistiche e letterarie: un quadro od un romanzo possono illuminare la visione del mondo altrettanto bene (anche se su un piano diverso) di un teorema o di una legge scientifica. Ho descritto in (Nardelli 2014) le fruttuose possibilità di collaborazione tra informatica e arte per l’acquisizione di conoscenza sul mondo.

Quando parlo di conoscenza scientifica parlo essenzialmente di quella basata sulla razionalità e sulla logica, al cui interno la conoscenza di tipo sperimentale e riproducibile (la cosiddetta “scienza galileiana”) ha un particolare rilievo.

Secondo Denning e Rosenbloom nella storia dell’umanità esistono tradizionalmente tre grandi domini della conoscenza scientifica. Uno è quello della *materia non vivente (physical)*, i cui oggetti di studio sono: materia, energia, gravità, moto, molecole, etc., e al cui interno sono comprese discipline quali la fisica e la chimica. Un altro è quello della *materia vivente (life)*, che studia oggetti quali: cellule, organismo, metabolismo, riproduzione, evoluzione, e così via, e che contiene le cosiddette scienze della vita, dalla biologia alla medicina. Il terzo è il *dominio dell’uomo e della società (social)*, che si occupa di: comportamento, cognizione, interazioni sociali, relazioni economiche, etc., tra le cui discipline ricordiamo la psicologia e l’economia.

In ognuno di questi domini vi sono innumerevoli trattati scientifici che investigano i vari oggetti e concetti di studio, e vi sono interi corsi di studio universitari. Tutti i domini condividono tre comuni caratteristiche:

1. si occupano di fenomeni caratteristici, specifici e pervasivi,
2. possiedono un insieme di strutture e processi concettuali altamente interrelati,
3. forniscono una prospettiva unica ed utile per la comprensione del mondo intorno a noi.

Osservo inoltre che ognuno di questi domini in un certo senso “si basa” sul precedente. Non ci può essere vita senza materia fisica (perlomeno in un contesto scientifico). Non c’è uomo né società senza materia vivente (per lo meno nel mondo di adesso: forse in un futuro

molto lontano ci sarà una società di macchine, e tra pochi decenni potremo forse arrivare a porci il problema della natura vivente o meno di sistemi non basati sul materiale organico e sulla chimica del carbonio). Ritengo importante sottolineare che questa gerarchia non implica una scala di valori, aspetto questo che rientra nell'ambito dell'etica e non della scienza. Ogni dominio, ed ogni disciplina scientifica al suo interno, ci permette di capire aspetti differenti di ciò che stiamo studiando. Il punto di vista fisico su una persona o su un fenomeno, è diverso e mette in luce aspetti diversi della stessa persona di quelli evidenziati dal punto di vista biologico, da quello psicologico o sociologico o cognitivo.

L'approccio fornito dall'informatica nello studio della realtà, basato sulla computazione<sup>1</sup>, secondo Denning e Rosenbloom non si colloca in nessuno di questi tre domini. Anche se processi di computazione si trovano nella realtà fisica (percorso minimo seguito da un raggio di luce nella riflessione), nella realtà vivente (la produzione di una proteina sulla base del DNA), nella realtà sociale (le reti sociali che crescono secondo leggi di potenza), la computazione in quanto tale non è un fenomeno della materia, né della vita, né di un ente sociale. D'altro canto però l'informatica possiede le tre caratteristiche specifiche di un dominio scientifico sopra descritte e può pertanto essere individuato come il quarto grande dominio: il *dominio computazionale*.

A questo punto sorge naturale la domanda: cos'è la computazione? Si tratta di una domanda fondamentale, ma ancora irrisolta, così come anche nel dominio della materia vivente non è definitivamente risolta la domanda "Cos'è la vita?".

Mentre le prime definizioni formali di computazione sono state legate al calcolo di funzioni matematiche, fatto comprensibile se si tiene presente che sono state elaborate intorno agli anni venti del secolo scorso, quando gli unici dispositivi di calcolo esistenti erano quelli di calcolo numerico, quelle più moderne prescindono completamente da queste, pur ovviamente essendo sempre basate sul formalismo matematico. Ad esempio, le definizioni originarie assumevano che gli ingressi della computazione fossero completamente stabiliti prima dell'inizio della computazione, che il programma per svolgere la computazione rimanesse sempre lo stesso, e che la computazione si arrestasse producendo un risultato in un tempo finito. Nelle più moderne formalizzazioni (van Leeuwen e Wiedermann 2012), elaborate

---

<sup>1</sup> Usiamo il termine "computazione" per sottolineare che stiamo considerando anche gli aspetti del calcolo simbolico e non solo il calcolo numerico.

tenendo presente le moderne tecnologie ed infrastrutture informatiche, queste assunzioni sono state rimosse.

Da un punto di vista più astratto ma più informale Denning (2010) definisce la computazione come “lo studio dei processi informativi, naturali o artificiali”, dove un “processo informativo” (o “processo di elaborazione dell’informazione”) è una sequenza di rappresentazioni generata dalla trasformazione di una rappresentazione in un’altra, trasformazione che avviene sotto il controllo di un’altra rappresentazione.

Vediamo quindi che l’approccio dell’informatica, pur essendo figlio della matematica, se ne è poi distaccato, fino a diventare una disciplina completamente autonoma. Ed in effetti il “calcolo” della matematica e la “computazione” dell’informatica costituiscono due differenti punti di vista. Lo spiegherò affrontando il tema della “risoluzione di problemi” (*problem solving*) che spesso viene invocato come una delle utili conseguenze dello studio dell’informatica. Si tratta di una visione semplicistica e ingenua: i matematici hanno risolto problemi da millenni e certamente lo studio della matematica allena alla risoluzione dei problemi.

Questo non vuol certo dire che informatica e matematica sono la stessa disciplina, ma che la loro differenza è altrove. La differenza che conta è che si ha una *soluzione matematica* ad un problema quando la soluzione è una *formula che definisce* la risposta al problema stesso. Si ha invece una soluzione *informatica* ad un problema quando la soluzione è un *processo che computa* la risposta. Possiamo definire un processo come un *algoritmo* espresso in un *linguaggio* che viene eseguito da un *automa*.

In aggiunta a questa differenza tra matematica e informatica (“formula che definisce la risposta” contro “processo che computa la risposta”) vi è anche l’importante differenza nella natura del destinatario dell’attività di risoluzione di problemi. Nel caso della matematica è un altro matematico, o in generale una persona, nel caso dell’informatica è un dispositivo l’esecutore. Questa differenza, enunciata in (Nardelli 2016a), ripresa in (Nardelli 2016b) ed approfondita in (Lodi et al. 2017) fa sì che l’informatica apporti, rispetto alla matematica, un mutamento radicale di punto di vista. Il cambiamento di paradigma concettuale è costituito dal passaggio dal risolvere i problemi al far risolvere i problemi. Questa importantissima transizione «*dal risolvere i problemi al far risolvere i problemi*» (Nardelli 2019) costituisce la «differenza che fa la differenza» (Bateson 1972) e colloca l’informatica come disciplina a sé stante tra le altre scienze.

Un esempio paradigmatico del modo di descrivere il mondo da parte dell'informatica (che viene sinteticamente chiamato "pensiero computazionale") è esemplificato da una sequenza <sup>2</sup> tratta dal film "Apollo 13". Gli astronauti si trovano in una situazione critica a bordo del modulo lunare, e devono essere guidati da terra per costruire un filtro per il sistema di purificazione dell'aria. In questo caso, gli esperti del centro di controllo devono ragionare come informatici per far sì che un esecutore sia in grado di comprendere ed eseguire le istruzioni che gli vengono fornite in modo che possa essere raggiunto l'obiettivo desiderato. Qui l'esecutore non è un computer (elettronico): gli astronauti devono "fare gli *automi*" che eseguono un *algoritmo* (quello per la costruzione del filtro - che loro non conoscono perché non sono ingegneri degli impianti di climatizzazione) che da terra viene loro inviato scritto in un linguaggio che devono essere in grado di comprendere e le cui azioni devono essere in grado di *attuare*.

Negli ultimi tempi si è parlato molto di "pensiero computazionale", nel mondo della scuola, alle volte in modo critico (Nardelli 2017) e alle volte come se si trattasse di una nuova disciplina: in (Lodi et al. 2017) abbiamo spiegato perché non si tratta di una nuova disciplina e qual è l'uso da fare di questo termine.


Qui mi preme ricordare che, dal momento che una rappresentazione informatica è in grado di modellare qualunque scenario, ecco che con l'informatica abbiamo la possibilità di costruire ed animare scenari nelle più svariate discipline. Come ho discusso più in dettaglio in (Nardelli 2017a; Nardelli 2017c)

la forza dell'informatica in un contesto educativo è costituita dalla capacità di concretizzare mondi virtuali, grazie alle sue possibilità apparentemente illimitate di elaborare simboli di qualunque tipo ed in qualunque modo. Perché niente è utile per apprendere un concetto come farnese esperienza concreta. Con l'informatica è possibile costruire qualunque mondo vogliamo esplorare e dargli senso, definire un insieme di regole e dar loro vita, animandole mediante un'esecuzione su di un appropriato dispositivo automatico. Con l'informatica è possibile costruire una simulazione di fenomeni fisici, biologici o sociali di cui si conoscano (anche in modo parziale ed approssimato) le leggi, osservare cosa accade e "cosa accadrebbe se...". Ed estendendo queste simulazioni è possibile costruire quelle "realtà virtuali" di cui stiamo vedendo l'incalzante diffusione ed alle quali è comunque necessario fare attenzione per le possibili conseguenze sulla vita reale.

Il pensiero computazionale, cioè quella modalità di pensiero che si acquisisce mediante lo studio e la pratica dell'informatica, può essere definito (Nardelli 2018) come "l'insieme dei processi mentali usati per costruire scenari e specificare i modi mediante i quali

---

<sup>2</sup> <https://www.youtube.com/watch?v=vNaNxwATJqY>


un agente che processa informazioni (esecutore) può operare in modo effettivo all'interno della situazione stessa per raggiungere uno o più obiettivi forniti dall'esterno". Esso possiede quindi non solo un'*utilità fondamentale*, perché forma gli studenti a quei concetti scientifici che sono alla base del mondo digitale in cui siamo ormai tutti completamente immersi, ma anche un'*utilità trasversale*, dal momento che permette di fornire un nuovo approccio all'acquisizione di conoscenza, quello fornito dalla lente **computazionale**  ovviamente, studiare i fenomeni naturali o artificiali come se fossero computazioni, non vuol dire che lo siano! Si ottengono nuovi utili punti di vista. Ne è un esempio lo studio dei sistemi viventi (p.es. una cellula), nella quale le nuove descrizioni fornite dagli informatici permettono di ottenere, insieme a quelle tradizionali della biologia ed a quelle più formalizzate di matematici e fisici, un quadro notevolmente più completo.

Discutevo questo importante valore pedagogico dell'informatica già in (Nardelli 2006), arguendo che il costringere lo scienziato a porsi in questa ottica costituisce l'enorme valore aggiunto della cultura dell'informatica (o del pensiero computazionale, direi oggi), e la motivazione della sua forte potenzialità di costituire un *ponte tra la cultura dell'esattezza e dell'oggettività e la cultura della sfumatura e della soggettività*, tra la sfera scientifica e quella artistico/letteraria, tra l'approccio digitale e quello analogico.

## 2 Che cos'è l'informatica?

---

### 2.1 La rappresentazione

Abbiamo quindi visto che l'informatica si occupa di far risolvere problemi ad un esecutore (denominato spesso anche "agente"), che in modo completamente meccanico ed automatico elabora rappresentazioni per produrre rappresentazioni. Ho usato volutamente il termine "rappresentazioni" e non "simboli" perché, anche se dal punto di vista umano quelle rappresentazioni sono appunto simbolo di qualcosa di significativo, per l'agente la rappresentazione non ha alcun significato, né possiede un significato – per l'agente – l'elaborazione che esse  esegue.

In informatica è usuale parlare delle rappresentazioni come dei "dati", anche se impropriamente si utilizza sovente il termine "informazione".

Da un punto di vista formale il termine “informazione” denota un dato la cui acquisizione da parte di un ricevente determina una riduzione della sua incertezza rispetto ad un qualche fenomeno. Un esempio tipico può essere la percentuale dei voti ottenuti da un partito in un’elezione, che è sempre un dato, ma che costituisce un’informazione solo che chi la riceve non ne è a conoscenza. Questa visione dell’informazione è quella nota come “teoria matematica dell’informazione”, dovuta a Claude Shannon. Tale teoria, escludendo del tutto il significato dell’informazione per il ricevente – significato che è una componente soggettiva e quindi non è formalizzabile in modo univoco – ha reso possibile, proprio in virtù della sua radicale oggettività, lo sviluppo di tutte le tecniche di rappresentazione e trasmissione dei dati che sono al centro dell’informatica e delle telecomunicazioni. Pertanto è opportuno utilizzare sempre il termine “dati” invece di “informazioni” quando se ne parla in riferimento all’agente meccanico che li elabora.

Le rappresentazioni possono essere classificate in due grandi categorie: analogica e digitale. La prima è quella che, fino a qualche decennio fa, è stata la più usata nella storia dell’umanità. L’esempio tipico è la posizione delle lancette in un quadrante o la lunghezza di un’ombra per indicare l’ora del giorno. La seconda – in base alla quale nello stesso esempio il tempo viene rappresentato mediante cifre – caratterizza ormai la società contemporanea, che proprio per questo viene denominata “società digitale”. Nella rappresentazione analogica c’è una proporzione, un’analogia. Quanto più la lancetta si è mossa dalla sua posizione iniziale o quanto più l’ombra è lunga, tanto maggiore è il tempo trascorso. Nella rappresentazione digitale ci sono una serie di segni arbitrariamente scelti, le cifre, ai quali arbitrariamente assegniamo un valore: il fatto che il segno “5” rappresenti cinque oggetti è una convenzione arbitraria: gli antichi romani usavano a tal scopo il segno “V”. Osserviamo però che tale rappresentazione delle quantità mediante cifre è stata utilizzata dall’umanità da millenni: l’usavano i Babilonesi per il calcolo delle orbite astronomiche, l’usavano gli Egizi per il calcolo delle superfici dei terreni, ognuno con il loro insieme di segni per rappresentare le quantità.

Anche l’elaborazione meccanica ed automatica delle rappresentazioni può essere realizzata in modo analogico oppure digitale. Le prime calcolatrici aritmetiche meccaniche eseguivano somme e sottrazioni mediante spostamenti di aste o ruote, cioè con una manipolazione analogica di dati analogici. I moderni computer, invece, elaborano in modo digitale rappresentazioni digitali, cioè costituite da cifre. In tal caso la somma di due valori, ad esempio, non è la lunghezza totale




delle due aste (ognuna della quali rappresenta un valore) messe in fila, ma è il risultato dell'addizione tra i due numeri ognuno dei quali è la rappresentazione digitale del numero. Le dieci cifre del nostro sistema di rappresentazione (denominato appunto “sistema decimale”) sono sostituite, all'interno dei computer, da un sistema binario (cioè che utilizza solo i due segni “0” e “1”), per semplice convenienza tecnologica. Per un moderno calcolatore basato sull'elettricità avere due soli valori da rappresentare costituisce una notevolissima semplificazione, che porta ad ottenere dispositivi di calcolo più piccoli e più veloci.

## 2.2 Automa e linguaggio

Attualmente, quindi, gli “esecutori” di riferimento quando parliamo di informatica sono degli esecutori che elaborano rappresentazioni digitali per produrre altre rappresentazioni digitali. Il termine tecnico con cui nell'informatica si denotano tali agenti è “automa”, parola greca che è la radice di termini quali “automatico” e “automatismo” che indicano appunto ciò che viene eseguito senza riflessione e senza coscienza.

La caratterizzazione precisa di cosa costituisce un automa è un aspetto fondamentale dell'informatica, su cui la ricerca non ha ancora terminato le proprie indagini. Senza entrare in dettagli tecnici, la definizione tradizionale, quella della cosiddetta “macchina di Turing”, non è più, infatti, adeguata per caratterizzare in modo teoricamente preciso i moderni computer, connessi in rete e dotati di programmi che si evolvono e non si arrestano mai. Questo ovviamente non impedisce alla tecnologia di progredire nella realizzazione di computer sempre più veloci e sofisticati. A tal proposito, è interessante osservare che il termine “computer” denotava, fino a prima della seconda guerra mondiale, le persone che facevano i calcoli. Solo con la costruzione dei primi calcolatori elettronici, avviatasi dopo il termine delle guerra, il termine ha smesso di indicare le persone e si usa ormai solo per le macchine di calcolo.

Definire l'automa di riferimento per le elaborazioni, cioè specificare l'elenco delle operazioni che esso è in grado di eseguire ed esporre che cosa viene realizzato da ogni operazione, è il primo passo, che va condotto insieme alla definizione del linguaggio che verrà usato per descrivere, di volta in volta, le specifiche elaborazioni che l'agente dovrà eseguire. La descrizione di una specifica elaborazione è usualmente chiamata “programma”, ed il linguaggio che viene usato per fornire le descrizioni all'automa viene per questo chiamato “linguag-

gio di programmazione”. In un certo senso si può dire che la definizione dell’automa e del suo linguaggio di programmazione sono due lati della stessa medaglia, sia perché non ha senso esporre  operazione per automa se non c’è un’istruzione nel relativo linguaggio che è in grado di invocarla, e simmetricamente perché non ha senso inserire nel linguaggio un’operazione che l’automa non è in grado di eseguire.

## 2.3 Algoritmo

L’automa e il linguaggio sono i due ingredienti concettuali di base: il terzo è costituito dall’algoritmo, cioè dalla versione astratta ed indipendente dallo specifico linguaggio di un’elaborazione che viene richiesta all’agente. Un *algoritmo* è un metodo risolutivo, che prescinde (cioè, astrae) da una specifica forma fisica di espressione. Tale astrazione è fondamentale dal punto di vista scientifico, perché fornisce generalità e ripetibilità alle varie elaborazioni. Si dice spesso che l’algoritmo è come una ricetta di cucina. Quest’analogia è vera nel senso che la descrizione di una ricetta è indipendente sia dagli specifici ingredienti che vengono effettivamente usati (la specifica di “100 grammi di farina doppio zero” e “300 grammi di marmellata di albicocche” non prescrive specifiche marche o varietà) sia da come avviene la loro effettiva manipolazione (“mescolare la farina e lo zucchero” non prescrive l’uso di mani o spatole). L’algoritmo viene poi espresso in un opportuno linguaggio, che nel caso di una ricetta di cucina può, ad esempio, essere l’italiano o il francese, e nel caso dell’informatica può essere Java o Python (due tra i linguaggi più diffusi al giorno d’oggi). Il *linguaggio* (nel caso dell’informatica – più esattamente – il “linguaggio di programmazione”) è il veicolo attraverso cui l’algoritmo assume una forma concreta e comprensibile da chi dovrà eseguirlo. L’analogia tra algoritmi e ricette continua parzialmente a reggere rispetto all’*automa* che esegue il programma, perché questi deve essere in grado, in entrambi i casi, di comprendere ed attuare le istruzioni espresse. Non è più corretta nel momento in cui osserviamo che l’agente che esegue una ricetta di cucina è un essere umano che certamente non realizza “meccanicamente” ed “automaticamente” le proprie azioni, ma utilizza la sua intelligenza e flessibilità per ottenere lo scopo descritto anche (in un certa misura che dipende dalla sua specifica esperienza) in presenza di errori nella ricetta o di variazioni nel contesto in cui opera. Diversamente da questo, un automa non ha alcuno spazio per interpretazione, adattamento e flessibilità. Anche una semplicissima differenza tra ciò che gli viene chiesto di fare è

ciò che è in grado di fare, intrinsecamente o nell'ambiente, determina l'impossibilità di eseguire con successo il programma.

I concetti di algoritmo, linguaggio, automa fanno parte del paradigma concettuale dell'informatica e possono essere impiegati con successo per fornire descrizioni complementari a quelle fornite da altre scienze, per esprimere il punto di vista informatico sui fenomeni, per descrivere come un informatico “vede il mondo”.

A questo punto uno potrebbe chiedersi: ma qual è la generalità degli automi che usa l'informatica? Se devo rappresentare un fenomeno biologico o astronomico ho necessità di automi diversi o no? La sorprendente risposta a questa domanda è che mi basta avere un solo automa, denominato “macchina universale di Turing”. Senza entrare in dettagli tecnici, tale automa è una particolare versione “flessibile” della macchina di Turing nel senso che si tratta di una macchina di Turing che riceve in ingresso sia la descrizione di una qualunque macchina di Turing sia la descrizione dell'ingresso che tale specifica macchina deve elaborare e, operando su tali due descrizioni, è in grado di simulare perfettamente il calcolo che la specifica macchina di Turing doveva eseguire.

Un qualunque computer, sia quelli di adesso che quelli giganteschi che occupavano intere stanze negli anni '50, non è niente altro che una macchina universale di Turing.

## 2.4 Dualità dei programmi

Riflettendo ulteriormente su questo aspetto e sull'esempio sopra descritto si capisce che la macchina universale di Turing esibisce un concetto molto potente dell'informatica, cioè la natura duale che hanno i dati rispetto all'esecutore. Infatti, alcuni dei dati che la macchina universale di Turing legge, e più precisamente quelli che descrivono la specifica macchina di Turing da simulare, sono considerati in un primo momento dall'esecutore come dei semplici dati, ma in un secondo momento costituiscono delle vere e proprie istruzioni che l'agente esegue. Questa natura duale dei dati, che possono essere interpretati anche come istruzioni, fornisce all'informatica un potere di “replicazione” che era stato posseduto prima solo da sistemi biologici.

Illustro questo potere di replicazione con un automa molto semplice. Descrivo prima il suo semplice repertorio di istruzioni e poi presento un semplice programma che è in grado di replicare sé stesso.

La prima istruzione del repertorio è un'istruzione di assegnazione del tipo

P=100;

che assegna a P il valore 100. Questo è solo un esempio di uso dell'istruzione, un altro potrebbe essere


Q=3;

che assegna a Q il valore 3. Abbiamo poi un'istruzione di stampa del tipo

stampa P;

che produce come risultato

100

(chiaramente  l'istruzione <<stampa Q;>> avrebbe invece prodotto <<3>>). L'automa possiede anche una variante di tale istruzione di stampa, cioè

stampa "P";

la cui esecuzione non produce più la stampa del valore eventualmente assegnato a P ma produce come risultato

P

(analogamente <<stampa "Q">> produrrebbe <<Q>>). Osserviamo a questo punto che l'istruzione di stampa può avere più oggetti da stampare, come per esempio accade con

stampa P Q;

che produce

100 3

e con

stampa "P=" P;

che invece produce

P = 100

Infine, l'automa possiede una *convenzione di ripetizione*, in base alla quale, ad esempio

3|X

costituisce un modo abbreviato di rappresentare

X; X; X;

Abbiamo pertanto che il programma

P=3|X; stampa P;

produce in uscita

X; X; X;

A questo punto il programma

P=3|X; stampa "P=" P;

produce in uscita

P = X; X; X;

Siamo pronti a questo punto a presentare e discutere il programma che replica sé stesso, che è costituito da queste due sole istruzioni

$P = 2$ |stampa "P=2" P; stampa "P=2|" P;

Osserviamo cosa fa l'automa quando esegue questo programma. Con la prima istruzione assegna a P – in base alla convenzione di ripetizione sopra descritta – il valore <<stampa "P=2" P; stampa "P=2" P;>> mentre con la seconda istruzione produce in uscita il valore <<P=2|>> seguito dal valore appena assegnato a P, cioè

$P = 2$ |stampa "P=2" P; stampa "P=2" P;

ovvero una copia esatta di sé stesso (lo spazio tra le due istruzioni è maggiore nel primo caso solo per motivi di chiarezza tipografica).


### 3 Dal coding al computational thinking per formare il cittadino digitale


**N**ella sezione precedente abbiamo visto alcuni esempi di programmi, cioè abbiamo fatto quello che oggi, usando un termine alla moda, si chiama *coding*, cioè programmazione. L'attività di coding non è, infatti, niente altro che la scrittura di programmi in un certo linguaggio di programmazione. Ed è chiaro che il suo utilizzo non è tanto finalizzato all'apprendimento di uno specifico linguaggio in sé, quanto a far capire la relazione che sussiste tra linguaggio e suo esecutore ed a far capire che un esecutore non è niente altro che un meccanismo automatico che, come un gigantesco orologio, ripete il percorso che noi abbiamo predisposto.

Presentare in modo appropriato il concetto di esecutore e del suo linguaggio è essenziale affinché i bambini acquisiscano sin da piccoli una reale consapevolezza di cosa sia l'informatica. Non è focalizzato sul padroneggiare la scrittura di programmi ma sul comprendere cosa vuol dire programmare un agente affinché esegua un certo compito. Questo percorso è per certi versi analogo a quello che si compie con la matematica con l'insegnamento della tavola pitagorica. Il che non è tanto che gli allievi sappiano a memoria che  $3 \times 2 = 6$  o  $12 : 4 = 3$ , quanto far loro capire che se 3 bambine hanno 2 biscotti ciascuna il numero totale di biscotti si ottiene con la moltiplicazione, mentre se 12 caramelle devono essere distribuite a 4 bambine il numero di caramelle per bambina si ottiene con la divisione. L'obiettivo non è quindi insegnare uno strumento operativo, quanto un metodo per la comprensione della realtà. In tal caso lo chiamiamo "pensiero matematico",

mentre nel caso dell'informatica lo chiamiamo “pensiero computazionale”, il termine con cui rendiamo in italiano l'espressione inglese “*computational thinking*”.

Sottolineo che l'uso di tale termine non implica che vada insegnata nella scuola una disciplina chiamata “pensiero computazionale”, come taluni sostengono. Sarebbe come sostenere che nella scuola la materia da insegnare sia il “pensiero matematico” e non la matematica. Ciò che deve essere insegnato nella scuola è l'informatica, e la sua pratica svilupperà nei ragazzi quella modalità di pensiero denominata “pensiero computazionale” che costituisce il particolare modo dell'informatica di descrivere il mondo in termini di processi automatici di manipolazione di rappresentazioni.

Il progetto “Programma il Futuro” (<http://programmmailfuturo.it>), realizzato nelle scuole italiane dal CINI (Consorzio Interuniversitario Nazionale per l'Informatica), ha appunto l'obiettivo di avvicinare i ragazzi a questa comprensione concettuale dell'informatica. Questo viene concretizzato attraverso una serie di attività, alcune delle quali sono di tipo tradizionale (cioè con carta e penna – <https://programmmailfuturo.it/come/lezioni-tradizionali>) mentre altre vengono eseguite al calcolatore (<https://programmmailfuturo.it/come/lezioni-tec-nologiche>). Le prime sono descritte fornendo un piano didattico per l'insegnante della lezione da realizzare, mentre per le seconde l'allievo è guidato passo passo alla comprensione dei vari concetti della programmazione mediante una serie di esercizi molto gradualmente e che partono da situazioni estremamente elementari (p.es., far muovere un personaggio di due caselle). Il progetto, in corso dall'anno scolastico 2014-15, ha avuto un'enorme diffusione nelle scuole, nel 7  delle quali c'è almeno un insegnante che partecipa. Tutti i dati sono accessibili a <https://programmmailfuturo.it/progetto/monitoraggio-del-progetto>.

A fianco di questa iniziativa operativa e pratica, il CINI – in collaborazione con insegnanti e pedagogisti – ha formulato una proposta per l'insegnamento dell'Informatica nel primo ciclo, che costituisce un progetto organico per l'inserimento di tale disciplina in quelle “Indicazioni Nazionali” del MIUR che **rappresenta**  documento di riferimento per la programmazione didattica. La proposta, consultabile alla pagina <http://consorzio-cini.it/gdl-informatica-scuola>, è costruita sulla visione che l'informatica è sia la disciplina scientifica di base che fornisce i concetti ed i linguaggi indispensabili per comprendere e per partecipare a pieno titolo alla società digitale, sia una disciplina di interesse trasversale che mette a disposizione un punto di vista ad-


dizionale, complementare a quello di altre discipline, per analizzare e affrontare situazioni e fenomeni.

Il disegno didattico – che arriva in realtà a comprendere anche il primo biennio delle superiori, coerentemente a quanto viene fatto negli altri paesi europei – è articolato su cinque ambiti che articolano l'intero percorso formativo:

- Algoritmi
- Programmazione
- Dati e informazione
- Consapevolezza digitale
- Creatività digitale

I primi tre ambiti sono quelli che sono stati delineati nel corso di questa sezione come quelli al centro dell'informatica, mentre gli ultimi due si occupano delle intersezioni tra l'informatica come disciplina scientifica ed i suoi utilizzi a scopo personale e sociale.

Come abbiamo accennato precedentemente, la rappresentazione digitale dei dati pervade sempre di più la nostra società: la quantità di dati raccolti ha raggiunto livelli incredibili. Una stima del 2017 riporta che ogni giorno vengono creati  $2,5 \times 10^{18}$  byte di dati (cioè 2,5 ExaByte di dati) e che il 90% di tutti i dati digitali esistenti al mondo sono stati creati tra il 2016 e il 2018. Il problema, come argomentato in (Corradini, Nardelli 2017) è che siamo immersi in un universo digitale per il quale non abbiamo organi di senso. La natura non ci ha dotato di sensori per “avere consapevolezza” del mondo digitale né viviamo in un ambiente che ci scherma da essi (come accade per il nostro pianeta la cui atmosfera ci protegge da tutta una serie di radiazioni elettromagnetiche altrimenti pericolose). Non avendo pertanto in modo naturale delle contromisure verso i pericoli insiti in questo universo, non c'è alternativa all'educazione, che deve iniziare fin dai primi anni.

La formazione digitale dei cittadini è essenziale e dev'essere basata sulle fondamenta scientifiche del mondo digitale. Si tratta dello stesso processo che ha portato nei due secoli passati ad introdurre nell'istruzione scolastica le varie materie scientifiche (fisica, chimica, biologia,  man mano che la società si trasformava da agricola ad industriale. Introdurre tali materie nella scuola è stato essenziale affinché i futuri cittadini potessero capire che dietro, ad esempio, un apparecchio televisivo non c'era una qualche magia ma solo principi scientifici implementati in una tecnologia efficacemente ingegnerizzata.

A tal proposito io ricordo sempre una citazione da Leonardo da Vinci che, nel Trattato della Pittura, scrive: “*Quelli che s'innamoran di pra-*

*tica senza scienza son come 'l nocchiere, ch'entra in navilio senza timone o bussola, che mai ha certezza dove si vada", dove il "navilio" indica l'imbarcazione, e prosegue sostenendo che la pratica deve essere sempre costruita a partire da una buona teoria, senza la quale non si fa bene nulla.*

Parlare, come troppo spesso accade al giorno d'oggi, di formare le competenze digitali negli studenti senza formare nella disciplina scientifica che ne è alla base rischia di essere estremamente controproducente. I critici obiettano che è pur vero che il mondo è ormai digitale, ma non tutti devono fare gli informatici. Anche se quest'ultima affermazione è vera, ci si dimentica però che l'obiettivo della scuola non è costruire scienziati o letterati, ma persone in grado di leggere e comprendere la società contemporanea. La necessità di insegnare informatica nella scuola deriva quindi dalla necessità di fornire a tutti i futuri cittadini gli strumenti culturali per poter capire e poter decidere con consapevolezza. In ultima analisi, quest'opera di formazione è cruciale affinché la società continui ad essere libera e democratica. Senza una reale conoscenza non si può infatti attuare una vera democrazia.

La sfida dei prossimi dieci o vent'anni sarà quindi quella di diffondere in tutta la popolazione una vera cultura informatica, che parta dall'alfabetizzazione tecnologica (cioè il saper usare gli strumenti digitali), prosegua con la conquista della cittadinanza digitale (cioè il conoscere le regole del corretto uso di tali strumenti nell'ambito delle relazioni sociali) e diffonda le basi scientifiche di tale disciplina (quindi il capire quei concetti di base che abbiamo precedentemente tratteggiato).

Su questa linea ci si sta muovendo in tutto il mondo evoluto ed anche a livello europeo Informatics Europe, l'associazione dei dipartimenti universitari di Informatica di cui sono attualmente presidente, ha formulato verso la Commissione Europea un invito ad inserire l'istruzione informatica per tutti tra le sue priorità (<http://informatics-europe.org/news/434-inf4all.html>).

L'informatica costituisce una vera e propria rivoluzione per l'umanità, che io chiamo la "terza rivoluzione dei rapporti di potere" prime due sono state quella della stampa e quella industriale.

L'invenzione nel quindicesimo secolo della stampa a caratteri mobili ha provocato una rivoluzione nella società sia di tipo tecnico, perché ha reso possibile produzione ti in modo più veloce e più economico, sia di tipo sociale, perché ha reso possibile una più diffusa circolazione della conoscenza. In ultima analisi, ciò che è accaduto è stata la *prima rivoluzione nei rapporti di potere*: l'autorità non era più legata




alla parola, al dover essere in un certo luogo in un certo momento per poter sapere ed apprendere dalla viva voce del maestro. La conoscenza rimane sempre un potere, ma questo potere non è più confinato alle persone che lo posseggono o a coloro che possono essere vicine ad esse nel tempo e nello spazio. La replicabilità del testo ha implicato la replicabilità della conoscenza in esso contenuta a distanza di tempo e di spazio. Tutti coloro che sapevano leggere potevano avere adesso accesso alla conoscenza. Questo ha messo in moto cambiamenti sociali epocali: la diffusione della conoscenza scientifica, giuridica e letteraria hanno dato un enorme impulso all'evoluzione della società, che è diventata sempre più democratica.

Nel giro di due secoli e mezzo, quasi ottocento milioni di libri stampati in Europa hanno messo in moto un irreversibile processo di evoluzione sociale. La conoscenza scientifica, rivoluzionata dal metodo galileiano, proprio grazie alla stampa si diffonde in tutta l'Europa e costituisce uno dei fattori abilitanti della successiva rivoluzione, quella industriale, che io individuo come la *seconda rivoluzione nei rapporti di potere*.

Questa, avviatasi nel Settecento è stata altrettanto dirompente: la disponibilità di macchine ha in questo caso reso replicabile il lavoro fisico delle persone. Le braccia umane non sono più necessarie perché la macchina opera al loro posto. Si ottiene una rivoluzione tecnica, perché si replicano più velocemente i manufatti. Le macchine possono produrre giorno e notte senza stancarsi, possono addirittura produrre altre macchine, amplificano e potenziano le capacità fisiche degli esseri umani. Si ottiene una rivoluzione sociale: si abbattano limitazioni fisiche al movimento e all'azione. Una singola persona può spostare enormi montagne di terra con una ruspa, spostarsi velocemente con un'automobile, parlare con chiunque nel mondo con un telefono. L'evoluzione ed il progresso della società umana vengono quindi ulteriormente accelerati grazie alla possibilità di produrre oggetti fisici più velocemente e più efficacemente, per non parlare delle conseguenze in termini di trasporto di persone e cose. Il potere che viene messo in discussione in questo caso è quello della natura: l'umanità assoggetta la natura e ne supera i limiti. Si possono attraversare velocemente i mari, solcare i cieli, imbrigliare acqua e fuoco, spostare montagne.

La rivoluzione della stampa aveva dato una marcia in più all'umanità sul piano immateriale dell'informazione, la rivoluzione industriale ha fatto altrettanto per la sfera materiale. Il mondo si popola di "artefatti fisici" (cioè macchine) che iniziano ad incidere in modo esteso ed approfondito sulla natura del pianeta.

Poi, a metà del Novecento, dopo circa ottocento miliardi di macchine, si avvia lentamente la *terza rivoluzione nei rapporti di potere*, quella dell'informatica. All'inizio sembra essere niente altro che una variante evoluta dell'automazione prodotta dalla rivoluzione industriale, ma dopo qualche decennio si comincia a capire che è molto di più di questo, perché incide sul piano cognitivo e non su quello fisico. Non si tratta più di replicare la conoscenza statica dei libri e la forza fisica di persone e animali, ma quella “conoscenza in azione” che è il vero motore dello sviluppo e del progresso.

Col termine “conoscenza in azione” intendo quel sapere che non è soltanto una rappresentazione statica di fatti e relazioni ma un processo dinamico e interattivo di elaborazione e di scambio dati tra soggetto e realtà. Grazie alla rivoluzione informatica, questa “conoscenza in azione” (“*actionable knowledge*”  ebbero in inglese, cioè conoscenza pronta ad essere messa in azione) viene riprodotta e diffusa sotto forma di programmi software, che possono poi essere adattati, combinati e modificati a seconda di specifiche esigenze locali. È cambiata la natura degli artefatti, delle macchine, che produciamo. Non sono più artefatti fisici, sono “*artefatti cognitivi dinamici*” (o *macchine della conoscenza*), azione congelata che viene sbloccata dalla sua esecuzione in un computer e genera conoscenza come risultato di tale esecuzione. La conoscenza statica dei libri diventa conoscenza dinamica nei programmi. Conoscenza in grado di produrre automaticamente, senza l'intervento umano, nuova conoscenza. Abbiamo una rivoluzione tecnica, cioè l'elaborazione più veloce dei dati, ma anche una rivoluzione sociale, cioè la generazione di nuova conoscenza. Il potere che viene scardinato, in questo caso, è quello dell'intelligenza umana. Certe attività cognitive non sono più dominio esclusivo dell'umanità: lo vediamo in tutta una serie di giochi da scacchiera (dama, scacchi, go, ...) un tempo unità di misura per l'intelligenza e nei quali ormai il computer batte regolarmente i campioni del mondo. Lo vediamo in tutta una serie di attività lavorative un tempo appannaggio esclusivo delle persone e nelle quali sono ormai abitualmente utilizzati i cosiddetti “bot”, sistemi informatici basati su tecniche di apprendimento e deduzione automatica (“*machine learning*” e “*artificial intelligence*”).

Ci sono però due caratteristiche distintive di queste “macchine della conoscenza” che le rendono differenti dal modo di operare dell'essere umano: la flessibilità e l'adattabilità. Le persone sono intrinsecamente in grado di apprendere ciò che non sanno (mentre le macchine della conoscenza possono apprendere solo ciò per cui sono state progettate)









ed hanno imparato, attraverso milioni di anni di evoluzione, ad adattarsi flessibilmente a cambiamenti imprevisti nell'ambiente (mentre le macchine della conoscenza possono – ancora una volta – adattarsi solo ai cambiamenti previsti).

Pertanto, mentre questi artefatti cognitivi dinamici basati sull'intelligenza artificiali sono certamente utili al progresso della società umana, e si diffonderanno sempre di più mentre le persone cambieranno il tipo di lavoro che fanno (come d'altro canto è accaduto in passato: nell'Ottocento più del 90% della forza-lavoro era impiegata nell'agricoltura, adesso è meno del 10%) è della massima importanza che ogni persona sia appropriatamente istruita e formata nelle basi concettuali della disciplina scientifica che rende possibile la costruzione di tali macchine.

Solo così l'umanità potrà continuare a dirigere e governare il proprio futuro.

## Riferimenti bibliografici

---

- net. <http://www.aicanet.it/storia-informatica/storia-dell-informatica-in-mondo-digitale>
- G. Bateson, *Form, Substance and Difference*, in *Steps to an Ecology of Mind*, University of Chicago Press, 1972.
- I. Corradini, E. Nardelli, *Sicurezza informatica: non vediamo e non vogliamo accettarlo*. <http://link-and-think.blogspot.it/2017/01/sicurezza-informatica-non-vediamo-e-non-vogliamo-accettarlo.html>, 2017.
- P.E. Ceruzzi, ria dell'informatica, Apogeo Editore, 2013.
- P. Denning, *The great principles of computing*, American Scientist, 98:369–372, 2010.
- P.J. Denning, P.S. Rosenbloom, *Computing: The fourth great domain of science*, Comm. ACM, 52(8), 27-29, 2009.
- G. Forsythe, *What to do till the computer scientist comes*, Math. Monthly 75:454-462, Amer, 1968.
- J. van Leeuwen, J. Wiedermann, *Computation as an unbounded process*, Theoretical Computer Science, 429, 202-212, 2012.
- M. Lodi, S. Martini, E. Nardelli, *Abbiamo davvero bisogno del pensiero computazionale?* Mondo Digitale, n. 72, AICA, 2017.
- E. Nardelli, *Informatica: un ponte tra le due culture*. In *Informatica: cultura e società*, Collana di Testi e Studi vol. 366. ISBN: 978-8-843039-36-4, Carocci editore, 2006 
- E.N elli, *A Viewpoint on the Computing-Art Dialogue: The Classification of Interactive Digital Artworks*. Leonardo, vol. 47, n. 1, pag  43-49, 2014.
- E. Nardelli (2016a), *Gli studenti dell'era digitale: automi o pensatori critici?* Aprile 2016. <https://www.ilfattoquotidiano.it/2016/04/21/gli-studenti-dellera-digitale-automi-o-pensatori-critici/2654546/>
- E. Nardelli (2016b), *Il pensiero computazionale fornisce un nuovo linguaggio per la descrizione del mondo*. Dicembre 2016. <http://link-and-think.blogspot.it/2016/12/pensiero-computazionale-nuovo-linguaggio-descrizione-del-mondo.htm> 
- E. Nardelli (2017a), *Di cosa parliamo quando parliamo di pensiero computazionale*. Gennaio 2017. <http://link-and-think.blogspot.it/2017/01/di-cosa-parliamo-quando-parliamo-di.html> 
- E. Nardelli (2017b), *Chi ha paura del pensiero computazionale (un anno dopo)*. Maggio 2017. <http://link-and-think.blogspot.it/2017/05/chi-ha-paura-del-pensiero-computazionale-un-anno-dopo.html>
- E. Nardelli (2017c), *Informatica nella scuola: disciplina fondamentale e trasversale, ovvero "di cosa parliamo quando parliamo di pensiero computazionale"*. Scienze e Ricerche Magazine, Aprile 2017, supplemento a Scienze e Ricerche n. 47, pp. 36-40. 

# **Parte terza**

## **L'apprendere...**

