



UNIVERSITÀ  
DI TRENTO

DIPARTIMENTO

FACOLTÀ DI GIURISPRUDENZA

lawtech

The Law and Technology Research Group

# Diritto dell'era digitale

**Lezione 3 – Hardware, software, algoritmi**  
Università di Trento – Facoltà di Giurisprudenza  
a.a. 2022-2023

Roberto Caso

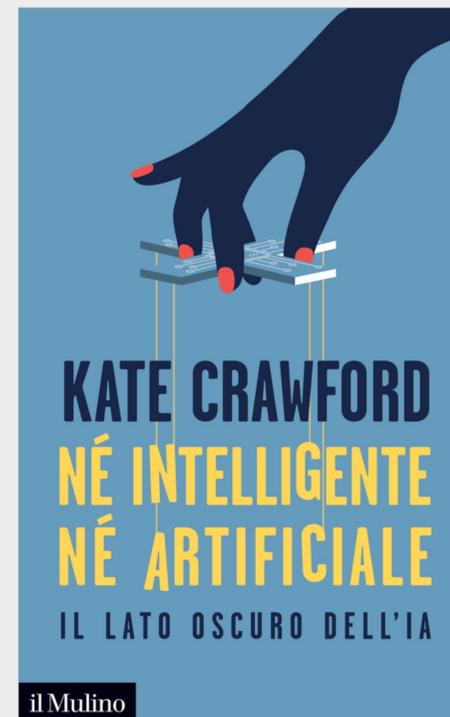
# Diritto e tecnologia

- «Esiste un rapporto molto stretto tra diritto e tecnologia. Più in particolare, esiste una relazione simbiotica tra il diritto e le attività umane che, sfruttando le acquisizioni della scienza, creano nuovi mezzi, strumenti, congegni, apparati atti a migliorare le condizioni di vita dell'uomo stesso [Rodotà, 1995]»

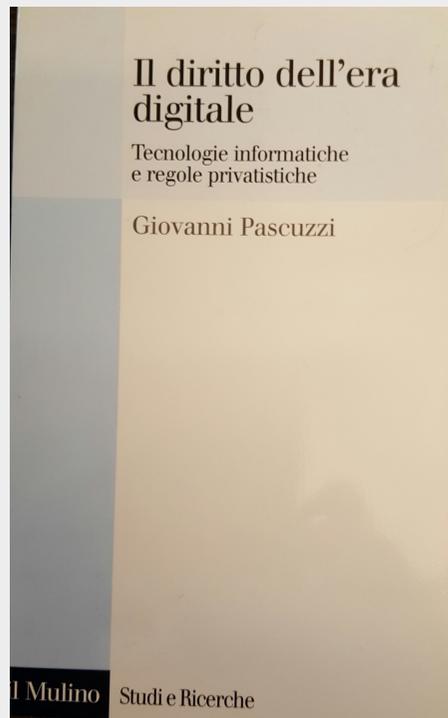


# Diritto, tecnologia e potere (ad es. IA)

- «[...] i sistemi di IA sono costruiti per interpretare il mondo e intervenire in esso in modo che sono principalmente vantaggiosi per gli stati, le istituzioni e le società di cui sono al servizio. In questo senso, i sistemi di IA sono espressioni di potere che discendono da forze economiche e politiche più ampie, creati per aumentare i profitti e centralizzare il controllo nelle mani di coloro che li detengono. Ma non è così che di solito viene raccontata la storia dell'intelligenza artificiale»



# Obiettivo del corso. Giovanni Pascuzzi (2002)



P. 16-17:

- «...verificare se e in che modo le tecnologie informatiche stanno cambiando le regole (operazionali) giuridiche ...»
- «cercare di capire se l'eventuale emersione di nuove regole in ragione dell'avvento delle tecnologie informatiche coincida con l'emersione di tratti caratteristici che possano indurre a parlare di: diritto dell'era digitale»

# Obiettivo del corso. Giovanni Pascuzzi (2020)



P. 21

- «Individuare i fenomeni che caratterizzano l'era digitale»
- «Analizzare i problemi posti dalla rivoluzione digitale»
- «...verificare se e in che modo le tecnologie informatiche stanno cambiando le regole (operazionali) giuridiche ...»
- «cercare di capire se l'eventuale emersione di nuove regole in ragione dell'avvento delle tecnologie informatiche coincida con l'emersione di tratti caratteristici che possano indurre a parlare di: diritto dell'era digitale»

# L'ordine del ragionamento

1. Era digitale
2. L'architettura del computer: cenni. Hardware/Software
3. Algoritmo. Software. Codice sorgente aperto/chiuso

# 1. Era digitale: concetti di base

- Digitale: segnale, misurazione o rappresentazione di un fenomeno attraverso numeri
- Scrittura binaria dei numeri: notazione posizionale in base 2
- *Binary digit* (bit)

# 1. Rappresentazione binaria e decimale di alcuni numeri

0000	=	0
0001	=	1
0010	=	2
0011	=	3
0100	=	4
0101	=	5
0110	=	6
0111	=	7
1000	=	8
1001	=	9
1010	=	10
1011	=	11
1100	=	12

# 1. Rappresentazione. Elaborazione. Comunicazione

- a) Rappresentazione. Rappresentare tutte le forme espressive in forma binaria.
- b) Elaborazione. Il codice binario può essere trattato con strumenti automatici.
- c) Comunicazione. Convergenza tra tecnologie informatiche e della comunicazione,

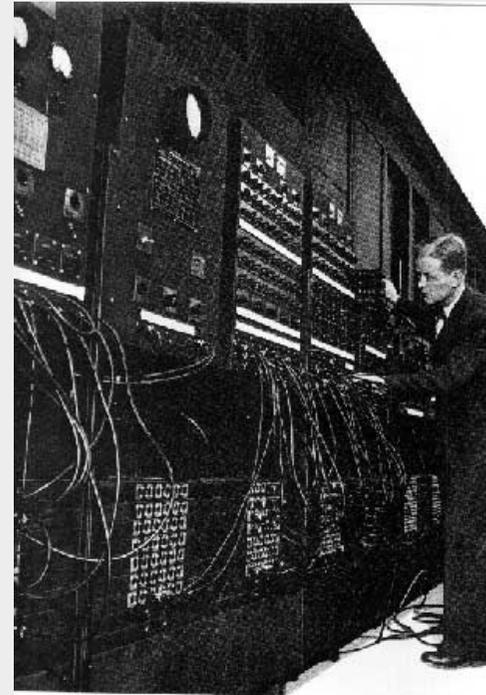


# 1. La macchina analitica (Babbage): ci sono già le idee di base del computer

- Un linguaggio semplificato che possa essere compreso dal calcolatore: codice binario
- Il codice binario è incorporato in dispositivi di ingresso che poi si chiameranno di **input**
- Il calcolatore (la macchina, **hardware**) ha un cuore che processa le informazioni (processore) e le restituisce elaborate (**output**)
- Una memoria temporanea per il “processamento” dei dati (**Random Access Memory** o **RAM**)

# 1. Gli impulsi elettrici ed i primi antesignani del computer moderno

- Atanasoff e Berry sviluppano tra 1939 ed il 1942 i primi calcolatori ad impulsi elettrici
- Mark I e Mark II
- Mauchly ed Eckert ultimano nel 1946 L' Electronic Numerical Integrator and Computer (ENIAC), raffigurato accanto (fotografia tratta da <http://www.eingang.org/Lecture/eniac.html>)



# 1. Hardware vs. Software?

- La distinzione è sottile (e, da alcuni punti di vista, questionabile)
- Nei primi antesignani dei computer il cambiamento di alcune funzioni richiedeva di intervenire fisicamente sulla macchina (ad esempio, modificando i circuiti, mettendo mano a spinotti etc.)
- Oggi queste funzioni vengono svolte da sistemi operativi (software)
- In linea teorica le funzioni svolte dal software possono essere tradotte in hardware
- **La natura ambigua del software: testo e macchina**

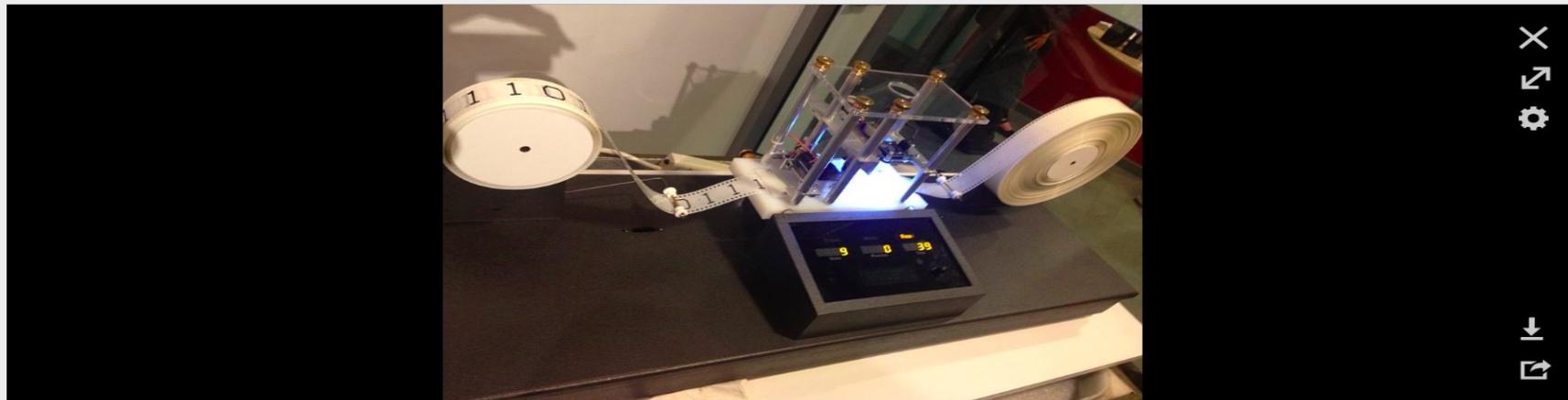
# 1. Una questione fondamentale

## Macchina/Testo

## 2. The imitation game (Alan Turing)



## 2. La macchina di Turing



Un esempio di macchina di Turing

 **GabrieleF** - Opera propria

 **Ulteriori dettagli**

 CC BY-SA 3.0

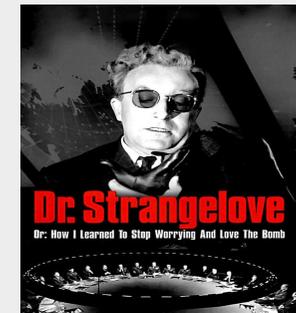
## 2. La macchina di Turing (Cappelli 2008)

- «Modello di agente di calcolo adatto a simulare la logica di qualsiasi algoritmo computazionale. La macchina formale fu proposta nel 1936 dal logico e matematico britannico Alan Turing, come sistema astratto che, opportunamente programmato, era capace di eseguire ogni tipo di operazione (l'idea di Turing era di rendere automatica una macchina da scrivere). Oggi ne esistono molte varianti, la più semplice delle quali è la macchina di Turing a nastro, formata da un'unità di controllo contenente un programma con un numero finito di istruzioni, da un nastro di lunghezza illimitata suddiviso in celle e da un'unità di lettura e scrittura sul nastro in grado di spostarsi avanti e indietro di un numero qualsiasi di celle e di leggere e scrivere in una qualsiasi delle celle un simbolo di un alfabeto prefissato. Questa macchina, pur nella sua semplicità, può calcolare in un numero finito di passi elementari qualsiasi funzione computabile. Il nastro si estende idealmente in modo infinito nei due versi e risulta diviso in celle, ciascuna contenente un simbolo appartenente a un insieme finito di simboli detto alfabeto».



## 2. John von Neumann

- John von Neumann (1903-1957): ideatore dell'architettura di Neumann
- Con von Neumann si inizia a delineare la struttura di un computer commercializzabile
- In cui il calcolatore ha una struttura (**architettura**) fissa in grado di svolgere differenti funzioni impartite da differenti software

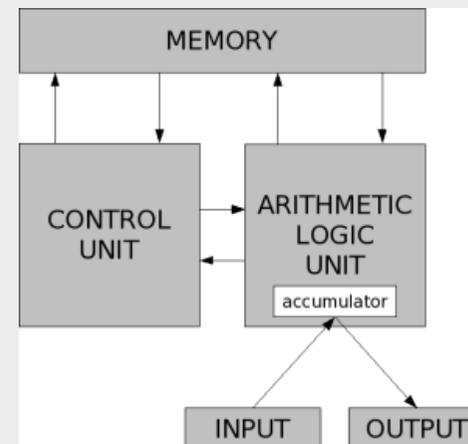


## 2. Architettura di Neumann

Tratto da Wikipedia

([http://it.wikipedia.org/wiki/Architettura\\_di\\_von\\_Neumann](http://it.wikipedia.org/wiki/Architettura_di_von_Neumann))

1. CPU o unità di lavoro che si divide a sua volta in:
  - Unità operativa, nella quale uno dei sottosistemi più rilevanti è l'ALU (Arithmetic Logic Unit)
  - Unità di controllo
2. Unità di memoria, intesa come memoria di lavoro o memoria principale (RAM, Random Access Memory)
3. Unità di input, tramite la quale i dati vengono inseriti nel calcolatore per essere elaborati
4. Unità di output, necessaria affinché i dati elaborati possano essere restituiti all'operatore



## 2. Un computer moderno: una semplificazione

- Central Processing Unit (CPU)
- Random Access Memory (RAM)
- Read Only Memory (ROM)
- Memoria fissa (Hard Disk)
- Memorie portatili: non più schede perforate ma dischi magnetici e ottici, nastri magnetici, flash memory
- Meccanismi di input ed output tra i quali le periferiche: non più schede perforate ma monitor, tastiera, mouse, stampante etc.

## 2. Hardware

- Definizione dello standard ISO/IEC IEEE 24765 – Systems and software engineering – Vocabulary
- Hardware: apparecchiature fisiche utilizzate per elaborare, archiviare o trasmettere programmi o dati informatici; ovvero: tutti i componenti fisici di un sistema informativo o parte di essi

### 3. Il software: una possibile generica definizione

- Un insieme di istruzioni destinate al computer con il fine di far svolgere a quest'ultimo determinate funzioni

# 3. Programma

- Definizione dello standard ISO/IEC IEEE 24765 – Systems and software engineering – Vocabulary
- Una combinazione di istruzioni per computer e definizioni di dati che consentono all'hardware del computer di eseguire funzioni computazionali o di controllo; ovvero un'unità sintattica conforme alle regole di un particolare linguaggio di programmazione e composta da dichiarazioni o istruzioni necessarie per una determinata funzione, attività o soluzione di un problema

# 3. Software

- Definizione dello standard ISO/IEC IEEE 24765 – Systems and software engineering – Vocabulary
- Programmi o parte di programmi, procedure, regole e documentazione associata di un sistema di elaborazione delle informazioni; ovvero: programmi per computer, procedure e possibilmente documentazione e dati associati di funzionamento di un sistema informatico; ovvero: programma o set di programmi utilizzati per far lavorare un computer

# 3. Algoritmo

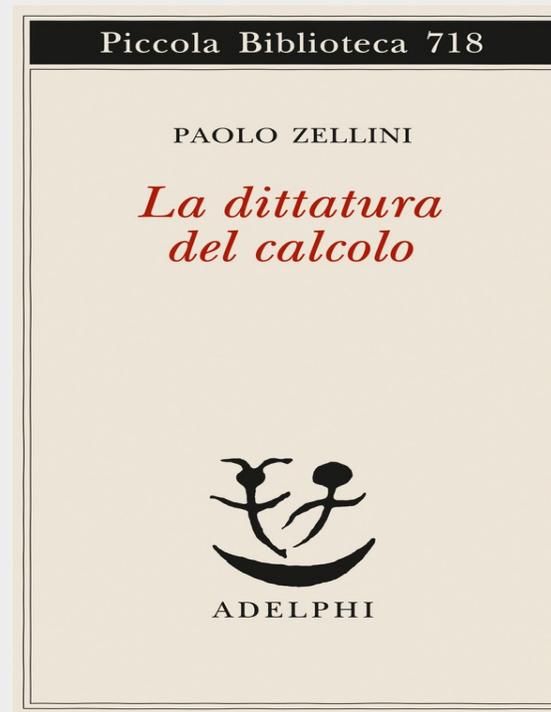
- Vocabolario Treccani:

“[...] qualunque schema o procedimento matematico di calcolo; più precisamente, un procedimento di calcolo esplicito e descrivibile con un numero finito di regole che conduce al risultato dopo un numero finito di operazioni, cioè di applicazioni delle regole”.

# 3. Algoritmo

- Definizione dello standard ISO/IEC IEEE 24765 – Systems and software engineering – Vocabulary
- Un insieme finito di regole ben definite per la soluzione di un problema in un numero finito di passaggi; ovvero: una sequenza di operazioni per eseguire un compito specifico; ovvero: un insieme ordinato e finito di regole ben definite per la soluzione di un problema

### 3. Paolo Zellini



## 3. Zellini

- “Un algoritmo consiste in una sequenza di istruzioni in base alle quali il calcolatore elabora un processo di calcolo”

## 3. Zellini

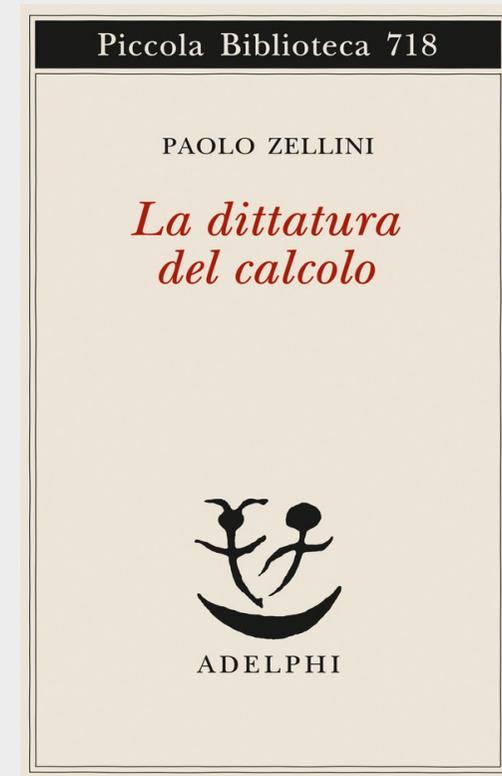
- “Gli algoritmi non fanno che estendere le funzioni di controllo e di ripartizione dei numeri in modi che possono diventare **inaccessibili, autoritari e categorici**: uno strumento utile alla società ma anche un rischio di sbilanciamento nel delicato rapporto fra categoricità e spontaneità, fra l'estrema imperiosità del meccanismo e la libertà di coscienza”

### 3. Sully: algoritmi, simulatori e fattore umano



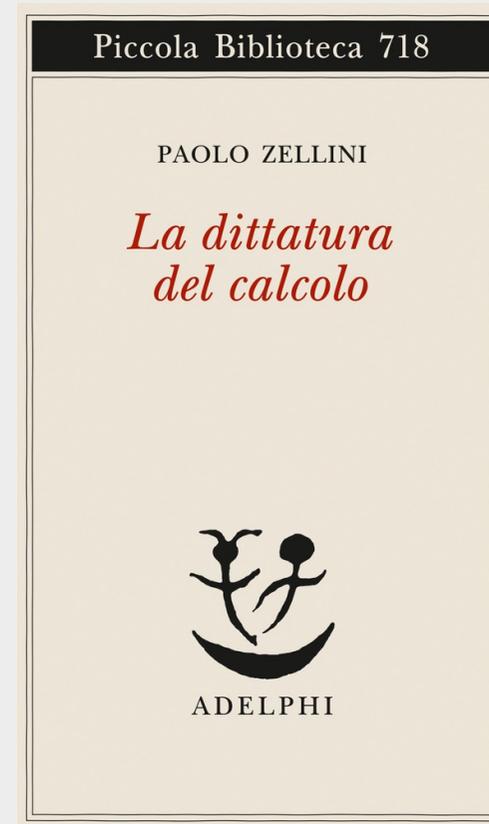
### 3. Zellini

- «Ma, al di là del trauma dell'incidente, qual è la ragione del tormento che affligge il generoso comandante? In fondo è pur sempre ancora sull'algoritmo che si basa il giudizio finale del National Transportation Safety Board sull'operato del pilota. Senza una simulazione convincente, aggiornata coi dati della reazione umana, il pilota non avrebbe modo di far valere le sue ragioni.»



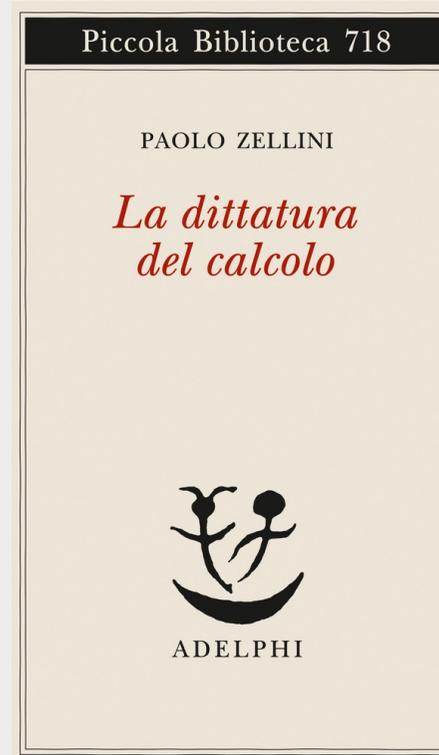
### 3. Zellini

«Si intravede allora, a questo punto, una virtuale iterazione, una ripetizione senza fine in cui si riconfigura ogni volta il divario tra realtà umana e realtà algoritmica. Infatti tutto ciò che fa l'algoritmo lascia fuori di sé una domanda sulla natura più intima dell'uomo e sullo stesso algoritmo. Si potrà definire una nuova procedura meccanica più perfezionata che ci aiuti a rispondere a tale questione, ma ci sarà sempre una domanda inevasa sul carattere precipuo della nostra identità e del nostro discernimento»

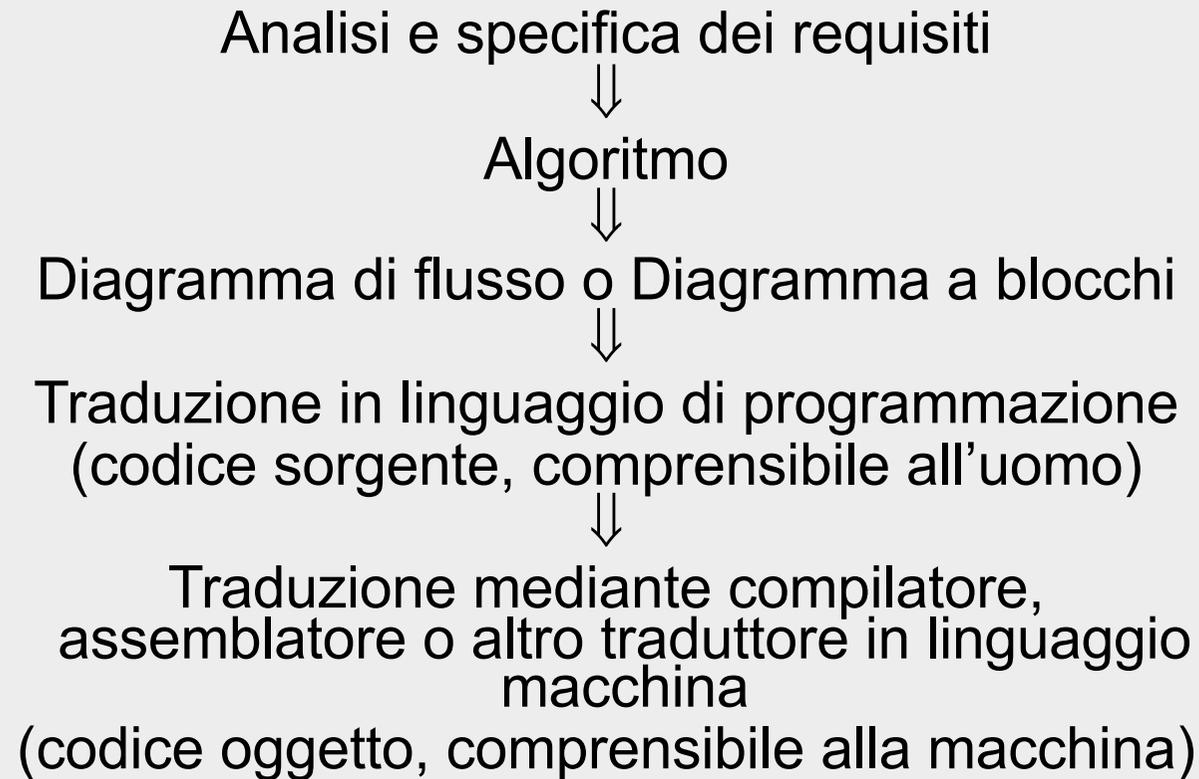


### 3. Zellini

- «Anche in presenza dei più perfezionati algoritmi si è obbligati a rimandare a qualcosa di esterno al loro meccanismo, a una responsabilità e a una libertà radicale che forse non esiste neppure, e che coincide infine con quella essenziale incompletezza che la tradizione filosofica e sapienziale, come pure le ricerche sulla natura della coscienza, hanno ontologicamente identificato come l'essenza stessa dell'uomo. Sullenberger, e con lui lo spettatore, si affacciano su questo precipizio, dove la matematica e la vita sembrano incontrarsi e confondersi.»



### 3. Il processo di progettazione e sviluppo: una semplificazione



# 3. La natura modulare del software

- Non si reinventa ogni volta la ruota! O forse sì?
- I software sono composti da moduli o subroutines
- I moduli “comunicano” tra loro (cioè sono resi compatibili tra loro) attraverso specifici moduli detti interfacce di programma (Application Program Interfaces o APIs)

# 3. Codice

- Definizione dello standard ISO/IEC IEEE 24765 – Systems and software engineering – Vocabulary
- Nell'ingegneria del software, istruzioni informatiche e definizioni dei dati espresse in un linguaggio di programmazione o in un formato prodotto da un assembler, compilatore o altro traduttore; ovvero: espressione di un programma per computer in un linguaggio di programmazione; ovvero: un carattere o un modello di bit a cui è assegnato un significato particolare

# 3. Codice sorgente

- Definizione dello standard ISO/IEC IEEE 24765 – Systems and software engineering – Vocabulary
- Istruzioni informatiche e definizioni dei dati espresse in una forma adatta per l'input a un assembler, compilatore, o altro traduttore

# 3. Codice oggetto

- Definizione dello standard ISO/IEC IEEE 24765 – Systems and software engineering – Vocabulary
- Istruzioni del computer e definizioni dei dati in un formato prodotto da un assembler o compilatore

### 3. Codice sorgente: un esempio

Esempio di codice sorgente in linguaggio C tratto da wikipedia:

```
#include <stdio.h>
int main(void)
{
printf("Ciao mondo!\n"); return 0;
}
```

### 3. Codice: sorgente (aperto) e oggetto

- Nel linguaggio informatico l'aggettivo “aperto” si riferisce alla pubblicità del codice sorgente
- Codice sorgente: istruzioni redatte in un linguaggio (di livello elevato) di programmazione comprensibile all'uomo (ad es., “C”, FORTRAN, COBOL, Pascal, BASIC, etc.)
- Codice oggetto: istruzioni in forma eseguibile ovvero in forma di linguaggio macchina comprensibile (solo) al computer (codice binario)

### 3. Una forma rudimentale di protezione tecnologica: la chiusura del codice

- La secretazione (chiusura) del codice sorgente costituisce una prima, rudimentale (ma potente) forma di protezione tecnologica

# 3. Decompilazione

- Complesso processo che consente a costi elevati di ricavare un codice sorgente simile (pseudo-codice sorgente) all' originale dal codice oggetto (l' unico disponibile). Viene detta anche più genericamente reverse engineering (ingegneria inversa)
- Microsoft Word: 27.000 linee di codice nel 1982 a 2.000.000 nel 2002 (fonte: Wikipedia.it)

# 3. Free Software Foudation

 **FREE SOFTWARE**  
FOUNDATION

[Log in](#) [Help!](#) [Members forum](#) [JOIN](#)  [NOW](#)

[about](#) [campaigns](#) [licensing](#) [membership](#) [resources](#) [community](#) [donate](#) [shop](#) [search](#)

---

**The Free Software Foundation (FSF) is a nonprofit with a worldwide mission to promote computer user freedom. We defend the rights of all software users.**

*Read this page in [Spanish](#) or in [Italian](#).*

As our society grows more dependent on computers, the software we run is of critical importance to securing the future of a free society. [Free software](#) is about having control over the technology we use in our homes, schools and businesses, where computers work for our individual and communal benefit, not for proprietary software companies or governments who might seek to restrict and monitor us. The Free Software Foundation **exclusively uses free software** to perform its work.

The Free Software Foundation is working to secure freedom for computer users by promoting the development and use of free (as in freedom) software and documentation—particularly the [GNU operating system](#)—and by [campaigning](#) against threats to computer user freedom like Digital Restrictions Management (DRM) and software patents.

**Our Core Work**

The FSF maintains historic articles covering [free software philosophy](#) and maintains the [Free Software Definition](#)—to show clearly what must be true about a particular software program for it to be considered free software.

The FSF sponsors the [GNU Project](#)—the ongoing effort to provide a complete operating system licensed as free software. We also fund and promote important free software development and provide development systems for GNU software maintainers, including full email and shell services and mailing lists. We are committed to furthering the development of the GNU Operating System and enabling volunteers to easily contribute to that work, including sponsoring [Savannah](#) the source code repository and center for free software development.

**The free software movement is one of the most successful social movements to arise from computing culture, driven by a worldwide community of ethical programmers dedicated to the cause of freedom and sharing. But the ultimate success of the free software movement depends upon teaching our friends, neighbors and work colleagues about the danger of not having software freedom, about the danger of a society losing control over its computing.**

### LibrePlanet 2023



[Register now for LibrePlanet 2023: Charting the Course.](#)

Submit a [lightning talk](#) at this year's LibrePlanet, either in person or online.

Submit a [Birds of a Feather \(BoF\) session](#) for this year's LibrePlanet.

### About the FSF

- [What is free software?](#)
- [FSF staff and board of directors](#)
- [The role of the FSF's board of directors](#)
- [Frequently asked questions](#)

# 3. <https://www.gnu.org/philosophy/philosophy.html>

## Filosofia del Progetto GNU

---

*Software libero* significa che agli utenti del software viene garantita la libertà, non si tratta di una questione riguardante il prezzo [NdT: in inglese *free* significa sia libero che gratuito, e quindi *free software* è ambiguo; in italiano il problema non esiste]. Abbiamo sviluppato il sistema operativo GNU per garantire agli utenti la possibilità di usare un computer in maniera libera.

Nello specifico, software libero significa che agli utenti vengono garantite le **quattro libertà fondamentali**: (0) di eseguire il programma come si desidera, per qualsiasi scopo, (1) di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità, (2) di ridistribuire copie in modo da aiutare il prossimo e (3) di migliorare il programma e distribuire pubblicamente i miglioramenti apportati.

Il software è diverso dagli oggetti materiali (come sedie, panini e benzina) in quanto può essere copiato e modificato molto più facilmente. Sono queste possibilità a rendere il software utile e crediamo che gli utenti di un programma – e non solo il suo sviluppatore – debbano essere liberi di trarre beneficio da queste.

*Per approfondimenti, si prega di selezionare una delle sezioni dal menù in alto.*

# Le quattro libertà fondamentali

Un programma è software libero se gli utenti del programma godono delle quattro libertà fondamentali:

- Libertà di eseguire il programma come si desidera, per qualsiasi scopo (**libertà 0**).
- Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (**libertà 1**). L'accesso al codice sorgente ne è un prerequisito.
- Libertà di ridistribuire copie in modo da aiutare gli altri (**libertà 2**).
- Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio (**libertà 3**). L'accesso al codice sorgente ne è un prerequisito

# 3. L'algoritmo

Vocabolario Treccani, prima accezione:

«[...] un procedimento di calcolo esplicito e descrivibile con un numero finito di regole che conduce al risultato dopo un numero finito di operazioni, cioè di applicazioni delle regole»

Pascuzzi 2020, 291:

«un insieme finito di regole ben definite per la soluzione di un problema in un numero finito di passaggi»

### 3. Algoritmi e processi decisionali automatizzati

Pascuzzi 2020, 291:

«L'elemento interessante è che, per molti, il concetto di algoritmo è divenuto sinonimo di automazione di un processo decisionale e, quindi, di decisione automatizzata»

# 3. Crittografia

Ad es.

- A chiavi asimmetriche (pubblica e privata)
- End to end

### 3. Crittografia: strumento di libertà e di controllo

- Anonimato e crittazione delle comunicazioni su Internet
- Diritto d'autore e Digital Rights Management (DRM) e Technological Protection Measures (TPMs)

# Roberto Caso

E-mail:

[roberto.caso@unitn.it](mailto:roberto.caso@unitn.it)

Web:

<http://www5.unitn.it/People/it/Web/Persona/PER0000633#INFO>

<http://lawtech.jus.unitn.it/>

<https://www.robortocaso.it/>

# Copyright

Copyright by Roberto Caso



Licenza Creative Commons

Quest'opera è distribuita con [Licenza Creative Commons  
Attribuzione - Condividi allo stesso modo 4.0 Internazionale](https://creativecommons.org/licenses/by-sa/4.0/)

La citazione di testi e la riproduzione di immagini costituisce esercizio dei diritti garantiti dagli art. 2, 21 e 33 Cost. e dall'art. 70 l. 1941/633