

# Istruzioni per un elaboratore digitale – Sviluppo software

# Istruzioni

La CPU esegue istruzioni in linguaggio macchina

Così come i dati, anche le istruzioni sono sequenze di 0 e 1. In linguaggio macchina un'istruzione si compone di

opcode

dati o indirizzi

Esempio 00000011 11000011

L'opcode identifica l'istruzione da eseguire sui dati o indirizzi di memoria

# Istruzioni

Un programma eseguibile da un elaboratore digitale è una sequenza di 0 e 1. Scrivere un programma complesso direttamente in linguaggio macchina è pressoché impossibile, quantomeno fortemente impratico.

Il linguaggio assembly, fornendo simboli per ognuna delle istruzioni in linguaggio macchina, rendendolo comprensibile ad un programmatore

00000011 11000011 -> add eax, ebx

Un programma chiamato assembler traduce poi il programma assembly in linguaggio macchina.

```
add eax, ebx  
mov al, [L1]  
mov eax, [L1]  
...
```

ASSEMBLER

```
00000011 11000011  
00011010 11001001  
01001000 10001001  
...
```

# Assembly

Margaret Hamilton



# Assembly

Sebbene la «vicinanza» dell'assembly al linguaggio macchina permetterebbe la creazione di programmi veloci ed efficienti, con accesso diretto all'hardware, l'assembly

- E' impratico
- Non è portabile (dipende dalla specifica architettura)

La maggior parte dei programmi è scritta in linguaggi (di programmazione) ad alto livello (di astrazione rispetto al linguaggio macchina).

# Linguaggi di programmazione

- Linguaggio macchina
  - Istruzioni sono sequenze di 0 e 1
  - Dipende dall'architettura dell'elaboratore
- Linguaggio assembly: livello di astrazione basso
  - Fornisce una corrispondenza quasi 1 a 1 con le istruzioni in linguaggio macchina
  - Dipende dall'architettura dell'elaboratore
- Linguaggi ad alto livello (es. C, C++, Python, Java...)
  - Fornisce costrutti più complessi delle istruzioni elementari eseguibili dall'elaboratore
  - Spesso indipendenti dall'architettura dell'elaboratore

# Es.: Assembly

```
%include "asm_io.inc"

segment .data

    msg db "Hello World!", 10, 0

segment .text

global ciaoasm
ciaoasm:

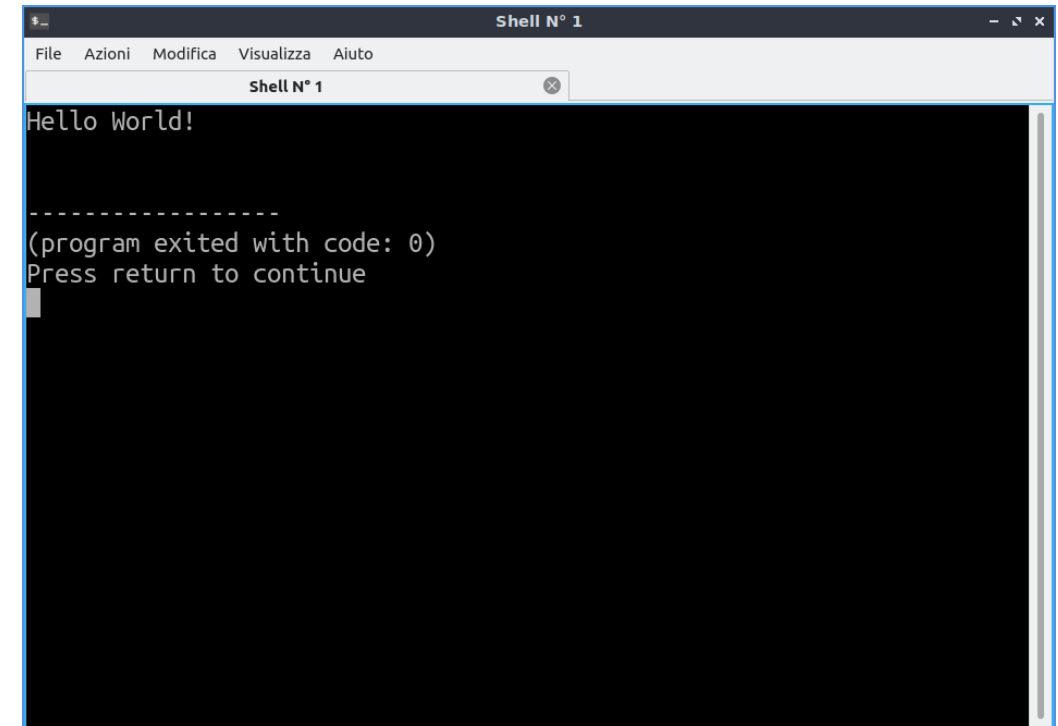
    enter 0,0
    pusha

    mov eax, msg
    call print_string

    popa

    mov eax, 0

    leave
    ret
```

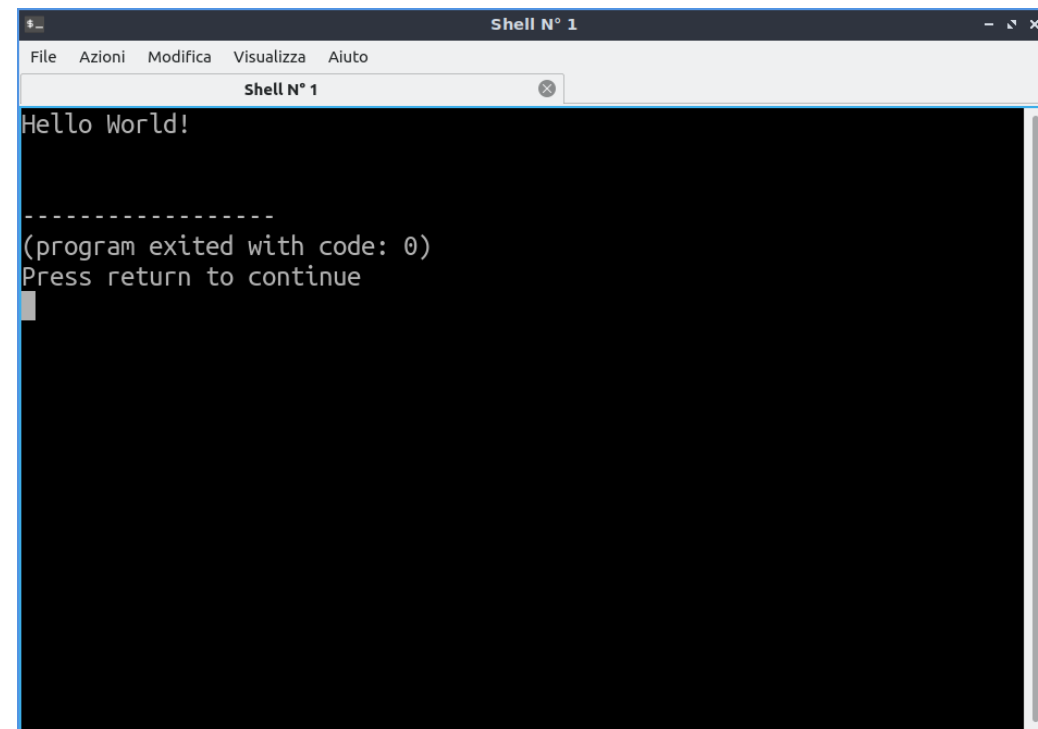


```
Shell N° 1
File Azioni Modifica Visualizza Aiuto
Shell N° 1
Hello World!
-----
(program exited with code: 0)
Press return to continue
```

## Es.: C++

```
#include <iostream>

int main() {
    std::cout << "Hello World!\n";
    return 0;
}
```



The screenshot shows a terminal window with a menu bar (File, Azioni, Modifica, Visualizza, Aiuto) and a tab labeled 'Shell N° 1'. The terminal output is as follows:

```
Hello World!

-----
(program exited with code: 0)
Press return to continue
```



# Scrittura di un programma

Conversione della descrizione accurata (specifiche) di un problema o di una classe di problemi in una sequenza di istruzioni per risolverlo mediante un elaboratore digitale.

Un linguaggio di programmazione di alto livello permette al programmatore di impartire all'elaboratore istruzioni complesse che «astraggono» il linguaggio macchina.

# Scrittura di un programma

Lo sviluppo di un programma si compone di diverse fasi: dalla raccolta delle specifiche del problema di risolvere, alla valutazione delle possibili alternative, fino alla scrittura del programma e ai test per verificarne il corretto funzionamento.

Semplice da usare non significa semplice da sviluppare!

# Fasi di sviluppo di un programma

1. Definizione (accurata) del problema da risolvere e delle operazioni necessarie per farlo
2. Progettazione del programma
3. Codifica in un linguaggio di programmazione
4. Test e debug
5. Scrittura della documentazione tecnica (es. manuale)



Ingegneria del software

# Ingegneria del software

## Motivazioni:

- Ridurre errori (i bug), realizzando verifiche già dalla fase di progettazione.
- Migliorare l'affidabilità.
- Documentare il codice, già fin dalle prime fasi di progettazione, migliorando il lavoro in team.
- Rendere modulare il software.

## Il primo «bug»

- 9 settembre 1947, Harvard University
- Il «Mark II» (23 t, la pianta occupava 370 mq) andava continuamente in errore
- Gli ingegneri aprono il Mark II per cercare di capire qual è la causa dell'errore...

# Il primo «bug»

9/9

0800 Antan started  
1000 " stopped - antan ✓

1300 (033) HP-MC { 1.2700 9.037 847 025  
2.130476415 9.037 846 795 conduct  
2.130476415 4.615925059(-2)  
(033) PRO 2 2.130476415  
conduct 2.130676415

Relays 6-2 in 033 failed special speed test  
in relay 11.00 test.

Relays changed

1100 Started Cosine Tape (Sine check)  
1525 Started Multi-Adder Test.

1545  Relay #70 Panel F  
(moth) in relay.

First actual case of bug being found.

1630 Antan started.  
1700 closed down.

Relay 2145  
Relay 2370

<https://education.nationalgeographic.org/resource/worlds-first-computer-bug>

## Affidabilità e «bug» - Ariane V (ESA)

- Ariane V: lanciatore dell'ESA (European Space Agency)  
[https://it.wikipedia.org/wiki/Ariane\\_5](https://it.wikipedia.org/wiki/Ariane_5)
- 4 giugno 1996, primo lancio (Ariane V Flight 501)

[https://www.youtube.com/watch?v=gp\\_D8r-2hww](https://www.youtube.com/watch?v=gp_D8r-2hww)

## Affidabilità e «bug» - Ariane V (ESA)

- 37 secondi dopo il lancio un dato di volo (la velocità orizzontale) in floating point a 64 bit viene convertito in un intero su 16 bit (che ammette valori -32768 a 32767), ma è troppo grande per essere rappresentato su 16 bit...
- I progettisti avevano riusato parte del codice relativo all'Ariane 4, dove la gestione di alcune eccezioni tra cui quelle relative alla velocità orizzontale erano state disabilitate («*physically limited or that there was a large margin of safety*»)
- Una reazione a catena a seguito dell'errore porta all'esplosione del razzo (costo del lancio: 370 milioni di dollari)

<https://medium.com/dataseries/crash-and-burn-a-short-story-of-ariane-5-flight-501-3a3c50e0e284>



# Affidabilità e «bug» - Mars Climate Orbiter **unIMC**

- 11 dicembre 1998: da Cape Canaveral, Florida, decolla il razzo Delta II 7425 con a bordo il *Mars Climate Orbiter*
- Obiettivo della missione scientifica è studiare il clima di Marte
- L'orbiter avrebbe dovuto essere posto in orbita intorno Marte ad una distanza di circa 140 km dalla superficie
- Il 23 settembre 1999 le comunicazioni si interrompono durante le operazioni di entrata in orbita
- La sonda si trovava a 57 km dalla superficie del pianeta anziché i 140-150 preventivati ed è andata distrutta dall'attrito con l'atmosfera

# Affidabilità e «bug» - Mars Climate Orbiter **unIMC**

- 11 dicembre 1998: da Cape Canaveral, Florida, decolla il razzo Delta II 7425 con a bordo il *Mars Climate Orbiter*
- Obiettivo della missione scientifica è studiare il clima di Marte
- L'orbiter avrebbe dovuto essere posto in orbita intorno Marte ad una distanza di circa 140 km dalla superficie
- Il 23 settembre 1999 le comunicazioni si interrompono durante le operazioni di entrata in orbita
- La sonda si trovava a 57 km dalla superficie del pianeta anziché i 140-150 preventivati ed è andata distrutta dall'attrito con l'atmosfera

# Affidabilità e «bug» - Mars Climate Orbiter **unIMC**

## Causa:

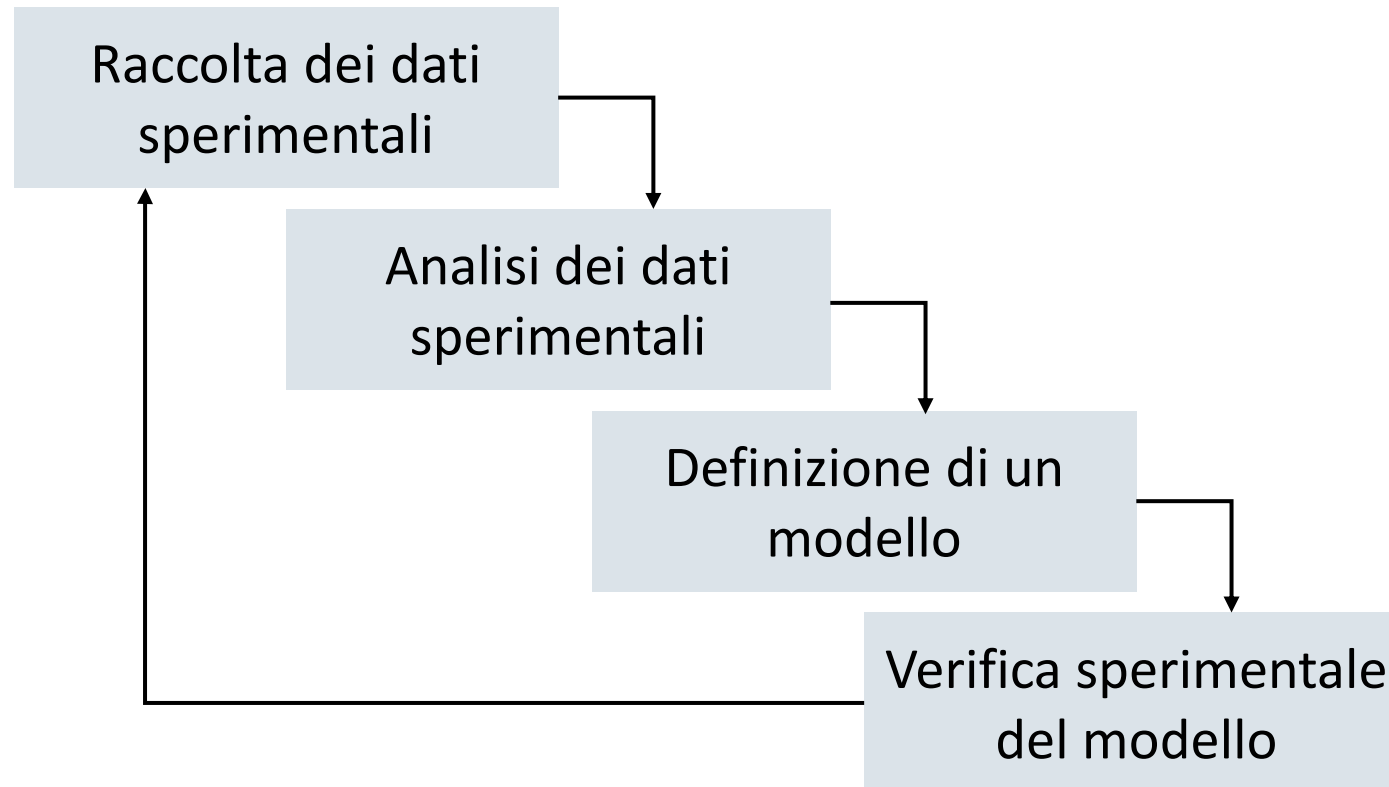
- Un software (di Lockheed Martin) usa il sistema imperiale britannico per le misure
- Un altro software (NASA), usato per una precedente missione, si aspettava risultati nel sistema internazionale di misura.

Costo della missione: 327.6 milioni di dollari.

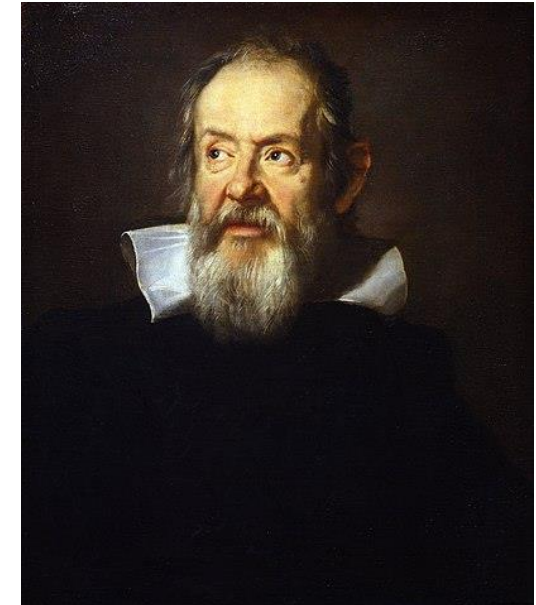
Altri bug «celebri»: <https://www5.in.tum.de/~huckle/bugse.html>

# Fasi di sviluppo di un programma

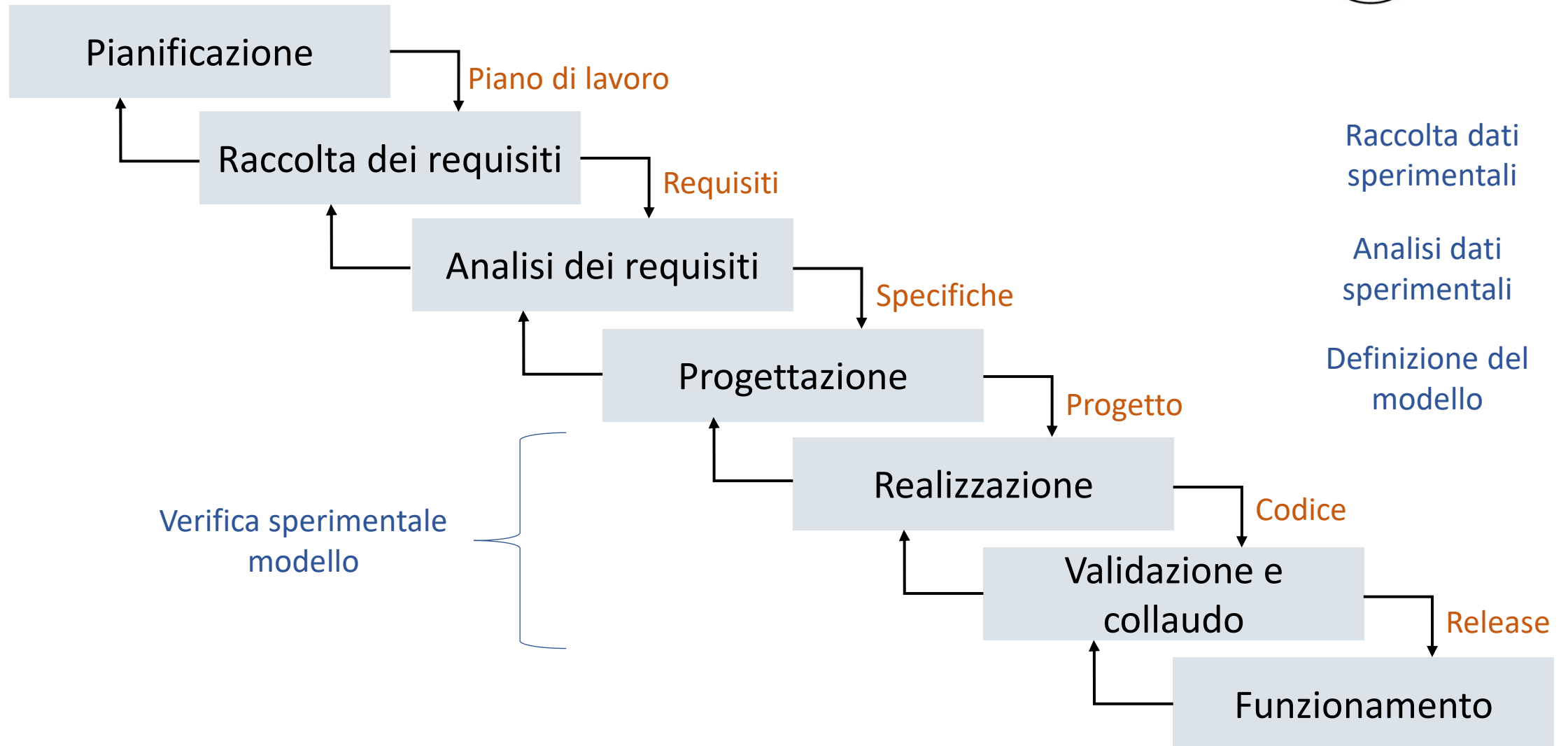
Esiste un parallelismo da sviluppo di un programma e metodo sperimentale.



Galileo Galilei



# Fasi di sviluppo di un programma



# 1. Pianificazione

- Stima di costi e rischi
- Definizione dei criteri di valutazione (finali, ma anche per monitorare i progressi in fase intermedia)
- Piano dettagliato per
  - Divisione dei compiti e responsabilità
  - Suddivisione delle risorse
  - Ispezioni e revisioni

## 2. Raccolta dei requisiti

- Definire chi fornirà i requisiti (gli «stakeholder»)
- Raccogliere i requisiti (es. attraverso «interviste»)
- Definire scenari d'uso, casistiche

### 3. Analisi dei requisiti

Definire che cosa deve fare il software (non il come). Dai requisiti si ottengono delle specifiche descritte «formalmente».

- Non è l'attività che si vuole automatizzare
- Non è uno specifico software
- Es. app per registrare video
  - Cosa: memorizzare sequenze di immagini digitali
  - Come: a 25 fps, a 60 fps, con MPEG-4, con H.264, ecc.



## 4. Progettazione - 5. Realizzazione

**Progettazione:** definizione del «come», cioè come il programma deve funzionare per soddisfare «che cosa» deve fare.

- Progettazione delle interfacce utente
- Progettazione della logica applicativa
- Progettazione dei dati

**Realizzazione:** traduzione del progetto in codice.

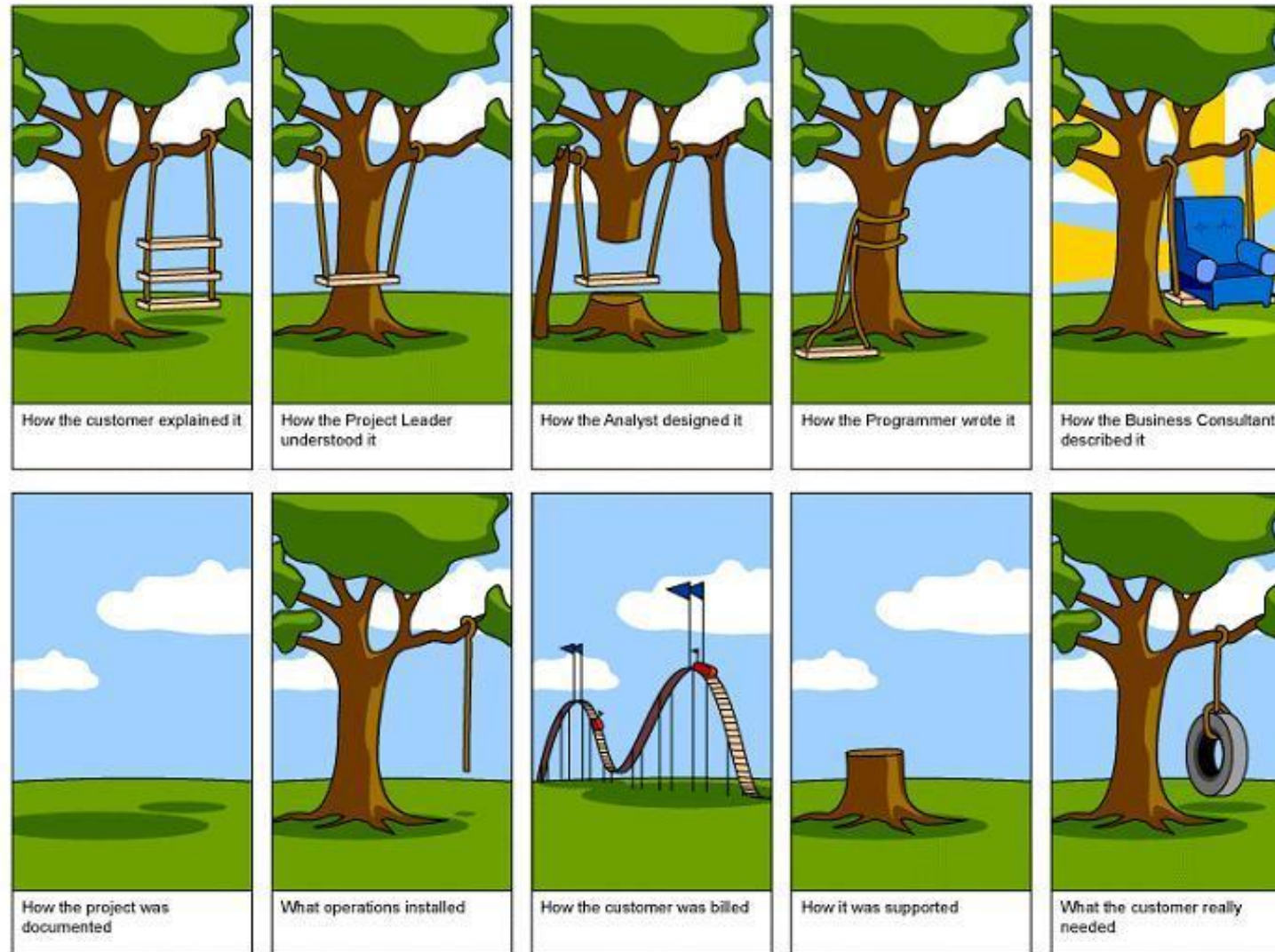
## 6. Collaudo – 7. Funzionamento

- Valutazione mediante metodi formali
- Test su casi d'uso
- Verifiche sulla base dei criteri di valutazione fissati in pianificazione
- Configurazione
- Documentazione



Rilascio (release)

# Fasi di sviluppo di un programma



## Programmi, linguaggi di programmazione e tutela del software

# Linguaggi di programmazione

**Codice sorgente:** programma scritto nel linguaggio di programmazione

**Codice eseguibile:** programma in linguaggio macchina effettivamente eseguibile dall'elaboratore digitale

# Linguaggi di programmazione

## Codice sorgente

- È un (o più) file di testo
- Scritto in un linguaggio leggibile da esseri umani
- Viene «tradotto» in linguaggio macchina da un software dedicato (es. assembler, compilatore, interprete)
- Indipendente dalla piattaforma

```
#include <iostream>

int main() {
    std::cout << "Hello World!";
    return 0;
}
```

# Linguaggi di programmazione

## Codice eseguibile

- È un (o più) file binario
- Contiene le istruzioni in linguaggio macchina che la CPU può eseguire
- Dipendente dalla piattaforma

```
7F 45 4C 46 02 01 01 00 00 00 00 00 00 00 00 00 00 03 00 3E 00 01 00 00 00 A0 10 00 00 00 00 00 00 40 00 00 00 00 00 58 3B 00 00 00
00 00 00 00 00 00 00 00 00 38 00 0D 00 40 00 00 1F 00 1E 00 06 00 00 00 04 00 00 00 40 00 00 00 00 00 00 00 00 00 40 00
00 00 00 00 00 00 D8 02 00 00 00 00 00 00 D8 02 00 00 00 00 00 00 08 00 00 00 00 00 00 00 03 00 00 00 04 00 00 00 18 03 00 00 00 00
00 18 03 00 00 00 00 00 00 18 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 90 07 00 00 00 00 00 00 90 07 00 00 00 00 00 00
10 00 00 00 00 00 00 00 00 01 00 00 00 05 00 00 00 10 00 00 00 00 00 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 A5 02 00 00 00 00 00 00 00 10 00 00 00 00 00 00 00 01 00 00 00 04 00 00 00 20 00 00 00 00 00 00 00 20 00 00 00 00 00 20 00
00 00 00 00 00 B0 01 00 00 00 00 00 00 B0 01 00 00 00 00 00 00 10 00 00 00 00 00 00 00 01 00 00 00 06 00 00 00 88 2D 00 00 00 00 00 00
88 3D 00 00 00 00 00 00 88 3D 00 00 00 00 00 00 88 02 00 00 00 00 00 00 D0 03 00 00 00 00 00 00 00 00 10 00 00 00 00 00 02 00 00 00 00 06
00 00 00 A0 2D 00 00 00 00 00 00 A0 3D 00 00 00 00 00 00 A0 3D 00 00 00 00 00 00 00 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 04 00 00 00 04 00 00 00 38 03 00 00 00 00 00 00 38 03 00 00 00 00 00 00 38 03 00 00 00 00 00 00 20 00 00 00 00 00 00 00
00 20 00 00 00 00 00 00 00 08 00 00 00 00 00 00 04 00 00 00 00 04 00 00 00 58 03 00 00 00 00 00 00 58 03 00 00 00 00 00 00 00 00 00 00
00 00 00 44 00 00 00 00 00 00 00 44 00 00 00 00 00 00 04 00 00 00 00 00 00 00 53 E5 74 64 04 00 00 00 38 03 00 00 00 00 00 00 00 00 00 38
03 00 00 00 00 00 38 03 00 00 00 00 00 20 00 00 00 00 00 00 20 00 00 00 00 00 00 00 00 00 08 00 00 00 00 00 00 00 50 E5 74 64 04 00
00 00 14 20 00 00 00 00 00 14 20 00 00 00 00 00 00 14 20 00 00 00 00 00 54 00 00 00 00 00 00 00 00 54 00 00 00 00 00 00 00 00 04 00
00 00 00 00 00 51 E5 74 64 06 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 10 00 00 00 00 00 00 52 E5 74 64 04 00 00 00 00 88 2D 00 00 00 00 00 00 88 3D 00 00 00 00 00 88 3D 00 00 00
00 00 00 78 02 00 00 00 00 00 78 02 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 2F 6C 69 62 36 34 2F 6C 64 2D 6C 69 6E 75 78 2D 78 38
36 2D 36 34 2E 73 6F 2E 32 00 00 00 00 04 00 00 00 10 00 00 00 05 00 00 00 47 4E 55 00 00 00 C0 04 00 00 00 03 00 00 00 00 00 00 00 00
00 04 00 00 00 14 00 00 00 03 00 00 00 47 4E 55 00 C0 56 A9 6D 51 B2 AC 7D 1E 20 E3 49 5D A3 5E 14 E9 0E 8D 60 04 00 00 00 10 00 00 00
01 00 00 00 47 4E 55 00 00 00 00 00 03 00 00 00 00 00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 91 00 01 00 00 00 09 00 00 00 00 00 00 00 D0 65 CE 6D 15 98 0C 43 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 D1 00 00 00 12 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 6D 00 00 00 00 12 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 A5 00 00 00 12 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 1F 00 00 00 20 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 ED 00 00 12 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 10 00 00 00 20 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 3B 00 00 00 20 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 00 00 00 12 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 DE 00 00 00 22 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 BD 00 00 11 00 1A
00 40 40 00 00 00 00 00 10 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
5F 5F 00 5F 49 54 4D 5F 64 65 72 65 67 69 73 74 65 72 54 4D 43 6C 6F 6E 65 54 61 62 6C 65 00 5F 49 54 4D 5F 72 65 67 69 73 74 65 72 54
4D 43 6C 6F 6E 65 54 61 62 6C 65 00 5F 5A 4E 53 74 38 69 6F 73 5F 62 61 73 65 34 49 6E 69 74 44 31 45 76 00 5F 5A 53 74 6C 73 49 53 74
31 31 63 68 61 72 5F 74 72 61 69 74 73 49 63 45 45 52 53 74 31 33 62 61 73 69 63 5F 6F 73 74 72 65 61 6D 49 63 54 5F 45 53 35 5F 50 4B
63 00 5F 5A 4E 53 74 38 69 6F 73 5F 62 61 73 65 34 49 6E 69 74 43 31 45 76 00 5F 5A 53 74 34 63 6F 75 74 00 6C 69 62 63 2E 73 6F 2E 36
00 5F 5F 63 78 61 5F 61 74 65 78 69 74 00 5F 5F 63 78 61 5F 66 69 6E 61 6C 69 7A 65 00 5F 5F 62 63 5F 73 74 61 72 74 5F 6D 61 69
6E 00 47 4C 49 42 43 58 58 5F 33 2E 34 00 47 4C 49 42 43 5F 32 2E 32 2E 35 00 00 00 00 02 00 03 00 03 00 00 00 02 00 00 00 00 00 03 00
```



|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| FF | 45 | 4C | 46 | 02 | 01 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 03 | 3E | 06 | 01 | 00 | 00 | A0 | 10 | 00 | 00 | 00 | 00 | 00 | 40 | 00 | 00 | 00 | 00 | 58 | 3B | 00 | 00 | 00 |
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 40 | 38 | 00 | 00 | 00 | 00 | 1F | 1E | 00 | 00 | 00 | 00 | 04 | 00 | 00 | 40 | 00 | 00 | 00 | 00 | 40 | 00 | 00 | 00 | 00 | 00 | 00 | 40 | 00 |    |
| 00 | 18 | 03 | 00 | 00 | 00 | 00 | 00 | 18 | 03 | 00 | 00 | 00 | 00 | 1C | 00 | 00 | 00 | 00 | 00 | 1C | 00 | 00 | 00 | 00 | 00 | 00 | 01 | 00 | 00 | 00 | 00 | 00 | 01 | 00 | 00 |    |
| 04 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 90 | 07 | 00 | 00 | 00 | 00 | 00 | 90 | 07 | 00 | 00 | 00 | 00 | 00 |    |
| 10 | 00 | 00 | 00 | 00 | 01 | 00 | 00 | 05 | 00 | 00 | 00 | 00 | 10 | 00 | 00 | 00 | 00 | 00 | 10 | 00 | 00 | 00 | 00 | 00 | 00 | 10 | 00 | 00 | 00 | 00 | A5 | 02 | 00 | 00 |    |    |
| 02 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |    |    |
| 00 | 00 | 00 | 00 | B0 | 01 | 00 | 00 | 00 | 00 | B0 | 01 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 10 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |    |    |
| 88 | 3D | 00 | 00 | 00 | 00 | 00 | 88 | 3D | 00 | 00 | 00 | 00 | 88 | 02 | 00 | 00 | 00 | 00 | 00 | 03 | 00 | 00 | 00 | 00 | 00 | 10 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |    |    |
| 00 | 00 | A0 | 2D | 00 | 00 | 00 | 00 | A0 | 3D | 00 | 00 | 00 | 00 | A0 | 3D | 00 | 00 | 00 | 00 | 00 | 00 | 02 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |    |    |
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |    |    |
| 00 | 20 | 00 | 00 | 00 | 00 | 00 | 08 | 00 | 00 | 00 | 00 | 00 | 04 | 00 | 00 | 00 | 00 | 00 | 00 | 03 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |    |    |
| 00 | 00 | 00 | 44 | 00 | 00 | 00 | 00 | 00 | 44 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 53 | E5 | 74 | 64 | 04 | 00 | 00 | 38 | 03 | 00 | 00 | 00 | 00 | 38 |    |
| 03 | 00 | 00 | 00 | 00 | 00 | 38 | 03 | 00 | 00 | 00 | 00 | 20 | 00 | 00 | 00 | 00 | 00 | 00 | 20 | 00 | 00 | 00 | 00 | 00 | 00 | 08 | 00 | 00 | 00 | 00 | 00 | 00 | 55 | 74 | 64 |    |
| 00 | 00 | 20 | 00 | 00 | 00 | 00 | 14 | 20 | 00 | 00 | 00 | 00 | 14 | 20 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |    |
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |    |
| 00 | 00 | 00 | 00 | 00 | 00 | 10 | 00 | 00 | 00 | 00 | 00 | 52 | E5 | 74 | 64 | 04 | 00 | 00 | 00 | 88 | 2D | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 88 | 3D | 00 | 00 |    |
| 00 | 00 | 78 | 02 | 00 | 00 | 00 |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |



• • •





# Sorgente



```
int main() {
    std::cout << "Hello World!";
    return 0;
}
```

# Reverse engineering

## «Decompilazione» del file eseguibile

- Risalire dal file eseguibile ad uno dei possibili codici sorgente.
- Per ragioni di interoperabilità con altri software, per studio e analisi, per copiare.

E' lecito il reverse engineering del file eseguibile?

Come è tutelato il software?

# Software – Copyright

- Il software è tutelato dal diritto d'autore, come un'opera letteraria
- Senza autorizzazione dell'autore non è possibile farne una copia, venderlo, modificarlo..
- Il proprietario del software è colui che ne detiene i diritti d'autore copyright
- Normalmente non si acquista il software, ma la licenza d'uso (eventualmente gratuitamente): la licenza stabilisce le facoltà dell'acquirente (es.: su quanti dispositivi il software può essere installato, per quanto tempo può essere utilizzato...)

# Software – Copyright

- Il copyright non protegge le funzionalità realizzate dal software, l'idea di base, ma solo la sua espressione
  - Chiunque potrebbe teoricamente analizzare un software e realizzarne un altro che svolga le stesse funzionalità, ma senza copiare il codice sorgente
  - Il software in quanto tale non è brevettabile. Codice della proprietà industriale, art. 45: [...] *Non sono considerate come invenzioni ai sensi del comma 1 in particolare: [...] i programmi per elaboratore;* (<https://www.normattiva.it/uri-res/N2Ls?urn:nir:stato:decreto.legislativo:2005-02-10;30>)
- Chi detiene i diritti d'autore può tenere nascosto il codice sorgente e fornire solo l'eseguibile. Questa è la «norma» per la maggior parte del software proprietario.

E' lecito il reverse engineering del file eseguibile?

In Europa (Direttiva 2009/24/CE) il reverse engineering è lecito senza autorizzazione dell'autore quando necessario per l'interoperabilità con un altro software, se:

- Chi lo attua ha acquistato almeno una licenza d'uso (per fare il reverse engineering è necessario avere almeno una copia del software)
- Le informazioni per l'interoperabilità non sono già disponibili
- Il reverse engineering è limitato solo alle parti necessarie per l'interoperabilità

## E' lecito il reverse engineering del file eseguibile?

### **Direttiva 2009/24/CE, Art. 6:**

*1. Per gli atti di riproduzione del codice e di traduzione della sua forma ai sensi dell'articolo 4, paragrafo 1, lettere a) e b), non è necessaria l'autorizzazione del titolare dei diritti qualora l'esecuzione di tali atti al fine di modificare la forma del codice sia indispensabile per ottenere le informazioni necessarie per conseguire l'interoperabilità con altri programmi di un programma per elaboratore creato autonomamente, purché sussistano le seguenti condizioni:*

- a) tali atti siano eseguiti dal licenziatario o da un'altra persona che abbia il diritto di utilizzare una copia del programma o, per loro conto, da una persona abilitata a tal fine;*
- b) le informazioni necessarie per ottenere l'interoperabilità non siano già facilmente e rapidamente accessibili alle persone indicate alla lettera a); e*
- c) gli atti in questione siano limitati alle parti del programma originale necessarie per conseguire l'interoperabilità.*

# Software – Copyright

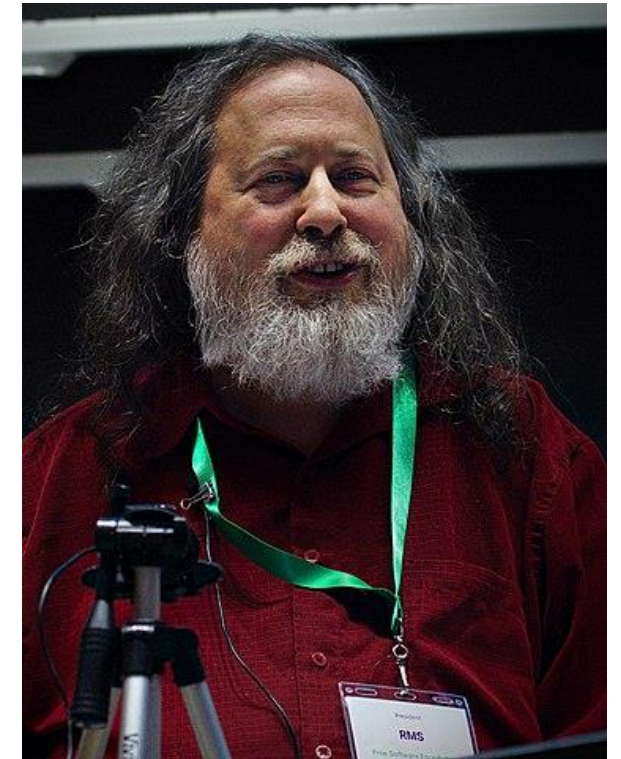
E' lecito il reverse engineering del file eseguibile?

- In Europa non è possibile vietare il reverse engineering nel contratto di licenza con l'utente finale.
- Negli Stati Uniti, per le opere protette da copyright, e dunque anche per il software, esiste il «Fair Use» che consentirebbe il reverse engineering per garantire l'interoperabilità o a fini di analisi. Tuttavia, il reverse engineering può essere vietato dalle condizioni del contratto di licenza.

# Free Software – Richard Stallman

- Programmatore e attivista del free software
- Nel 1985, fonda la Free Software Foundation (FSF)
- Autore di licenze per Free Software (es. GNU GPL – GNU General Public License)
- Free software (*software libero*): il software dovrebbe essere distribuito per dare agli utenti la libertà di usarlo, studiarlo, modificarlo e redistribuirlo

Richard Stallman





## Free Software – Aneddoto

- 1980, Laboratorio di Intelligenza Artificiale del MIT: Stallman vi lavora come ricercatore, programmatore, hacker
- La stampante, una Xerox 9700, non forniva informazioni sulle code di stampa
- Sembra che Stallman, che aveva già modificato il driver di una precedente stampante in tal senso, chiese alla Xerox il sorgente del driver per aggiungere (gratuitamente) funzionalità e che l'azienda rifiutò
- Essere incapaci di apportare quella modifica era un problema, considerando che molti utenti stavano anche ad un piano diverso rispetto la stampante...

# Free Software

- Open source: software rilasciato insieme al codice sorgente. Chiunque può studiarlo, modificarlo e redistribuirlo (anche a pagamento)
- Free Software: software rilasciato insieme al codice sorgente. Chiunque può studiarlo, modificarlo, redistribuirlo (anche a pagamento), ma il software deve essere distribuito mantenendolo libero, *senza assoggettarlo a restrizioni* (copyleft)

Free software  $\neq$  Open Source

# Free Software – Libertà fondamentali

*«Free software is a matter of liberty, not price. To understand the concept, you should think of 'free' as in 'free speech', not as in 'free beer'»*

*Richard Stallman*

1. Libertà di eseguire il programma per qualsiasi scopo
2. Libertà di studiare il programma e modificarlo
3. Libertà di «copiare» il programma in modo da aiutare il prossimo
4. Libertà di migliorare il programma e di distribuirne pubblicamente i miglioramenti, in modo tale che tutta la comunità ne tragga beneficio

# Licenze

- GNU GPL v3
- Apache v2.0
- BSD originale (no copyleft)
- MIT License (no copyleft)
- ...

<https://www.gnu.org/licenses>

# Licenze – GNU GPL v3

## Libertà di

- usare il software per qualunque scopo,
- modificare il software per adattarlo ai propri bisogni,
- condividere il software con gli amici e i vicini, e
- condividere le modifiche effettuate

Guida rapida: <https://www.gnu.org/licenses/quick-guide-gplv3.it.html>

Testo completo: <https://www.gnu.org/licenses/gpl-3.0.html>

# Open Source Initiative



**Open Source Initiative**

- Punto di vista pragmatico più che ideologico
- Promuove l'uso dell'open source anche nelle applicazioni commerciali
- Mantiene una lista di licenze software consistenti con i principi dell'open source (meno restrittivi di quelli del free software)

<https://opensource.org/licenses>

# Ricadute sociali

Ricadute di free software e open source sulla «società digitale»

- Il codice sorgente viene diffuso
- Chiunque (con le adeguate competenze) può modificarlo per migliorarlo o derivare altro software, non necessariamente a titolo gratuito, alle condizioni della specifica licenza (free software ≠ open source)
- L'autore mantiene i diritti

# Free software e open source

## Vantaggi:

- Trasparenza per gli utenti
- Interoperabilità, retrocompatibilità ed efficienza
- Elevati standard di sicurezza
- Libertà tecnica e indipendenza dal vendor
- Nuovi modelli di business



# Free software e open source

## Critiche:

- Licenza non economicamente sostenibile
- Non utile per categorie di software di nicchia
- Sviluppo a struttura «anarchica»

# Free software e open source

Modelli di business:

- Personalizzazione
- Consulenza
- Assistenza

## Esempi «celebri»

