

RiBau: il CANE torna a correre, un calcolatore didattico del 1970

Alessandro Cignoni, Giovanni A. Cignoni, Giuliano Pacini, Daniele Ronco

Sommario

Il *Calcolatore Automatico Numerico Educativo* (CANE) era un calcolatore didattico, esemplare pisano di una categoria con altri illustri esponenti. Era usato nel primo corso di laurea in informatica istituito in Italia, attivato a Pisa nel 1969/70. Il CANE era uno strumento didattico realizzato dagli studenti, fu infatti il risultato di due tesi. Il relatore fu Antonio Grasselli, uno dei principali promotori del corso di laurea. La ricostruzione del CANE è il contributo di Progetto HMR al 50° di Scienze dell'Informazione.

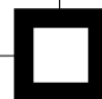
Abstract

A number of studies report about students' difficulties with basic flow-control constructs, and specifically with iteration. As part of a project whose long-run goal is identifying methodological tools to improve the learning of iteration constructs, we analyzed the answers of a sample of 164 high school students to three small programming tasks and two questions on their perception of difficulty. The results of the analysis suggest that more teaching efforts should be addressed to the development of a method to approach programming tasks and, more specifically for iteration, to the treatment of loop conditions in connection with the specifications in the application domain

Keywords: University; Computing History; Simulation; Computer Architecture; Machine Code

1. Introduzione

Il *Calcolatore Automatico Numerico Educativo* (CANE) si studiava al corso di *Teoria e Applicazioni delle Macchine Calcolatrici* al corso di laurea in *Scienze dell'Informazione*, il primo dedicato all'informatica attivato in Italia, all'Università di Pisa nel 1969/70. Fra gli studenti era popolare un'interpretazione alternativa



delle ultime due lettere dell'acronimo: "Non Esistente". Il CANE infatti era, diremmo oggi, una macchina virtuale: si usava facendo girare il *SimulCANE* sull'*IBM 7090* del CNUCE, il Centro Nazionale Universitario di Calcolo Elettronico istituito dall'Università di Pisa nel 1964.

Ricostruire vecchi calcolatori usando la simulazione software, cioè studiare e mostrare l'informatica di ieri con le tecnologie di oggi è una caratteristica di Progetto HMR [1]. Le precedenti esperienze nella ricostruzione della prima Calcolatrice Elettronica Pisana (CEP), sia secondo il primo progetto del 1956 [2] sia nella versione effettivamente realizzata nel 1957 [3], hanno simulato calcolatori che nella realtà erano macchine vere. *RiBau*, il progetto dedicato al CANE, ha invece realizzato una versione moderna di qualcosa che, già in origine, era un simulatore. Il valore aggiunto è la fruibilità: *SimulCANE* girava su un mainframe al quale gli studenti accedevano soltanto per interposta persona, attraverso gli operatori del CNUCE. Il simulatore del CANE di 50 anni dopo è un'applicazione web accessibile a tutti, a distanza, in ogni momento.

L'articolo racconta la storia del CANE e presenta la sua nuova versione web. È descritto il contesto storico, sia come riferimenti ad altri celebri calcolatori didattici del tempo, sia come uso nel corso di laurea in Scienze dell'Informazione. È discussa l'architettura del CANE, come funzionava l'originale e le scelte di ricostruzione. Infine, è descritta una breve sessione sul simulatore web – l'immane *Ciao Mondo*.

2. Calcolatori didattici, omini, leoni e pistoleri

Il CANE non fu l'unico calcolatore didattico "non esistente". L'idea di macchine esempio, pensate per spiegare, era già stata percorsa. Erano macchine usate per presentare agli studenti le basi dell'architettura dei calcolatori e della programmazione. Ripulite dalle complicazioni tipiche dei calcolatori reali, erano macchine ideali, sia nel senso platonico di modelli rappresentativi, sia per le esigenze pragmatiche della didattica.

Il CANE fu realizzato come tesi di laurea [4, 5] nel 1969/70, in pratica contemporaneamente all'attivazione del corso di laurea in Scienze dell'Informazione [6], e fu poi usato nelle lezioni di TAMC negli anni successivi. Al tempo c'erano almeno altri due esempi di calcolatori didattici: il *Little Man Computer* (LMC) [7] proposto nel 1965 da Stuart Madnick e John Donovan e usato nei corsi al Massachusetts Institute of Technology, e il *MIX* di Donald Knuth, pubblicato nel 1968 nel primo volume [8] del suo monumentale "The Art of Computer Programming".

Il nome del LMC di Madnick e Donovan non era scelto a caso. Nell'intento didattico di spiegare il funzionamento di una macchina a programma memorizzato, i due fecero ricorso a un omino che prelevava dalla memoria le istruzioni, svolgeva le operazioni logiche e aritmetiche necessarie per eseguirle, aggiornava gli stati dei registri e della memoria e poi ricominciava il suo ciclo di lavoro. LMC ha una lunga storia didattica, oltre i corsi al MIT fu adottato in molti testi, fra i più longevi c'è [9], riedito fino al 2014.

Anche il CANE aveva i suoi omini. La dispensa di Grasselli del 1972 [10] era illustrata da Alberto Fremura, pittore e vignettista labronico. Le pagine introduttive descrivevano le operazioni interne di un calcolatore come svolte da impiegati, fattorini e meccanici, metafora poi ripresa, con tanto di plastico, nelle lezioni che Grasselli tenne fra il novembre 1974 e il febbraio 1975 sulla RAI – di questi tempi un esempio particolarmente curioso di didattica a distanza [11]. Di Fremura è anche il leone che rappresentava il prestante IBM 7090 intento a eseguire il programma SimulCANE (fig. 1).

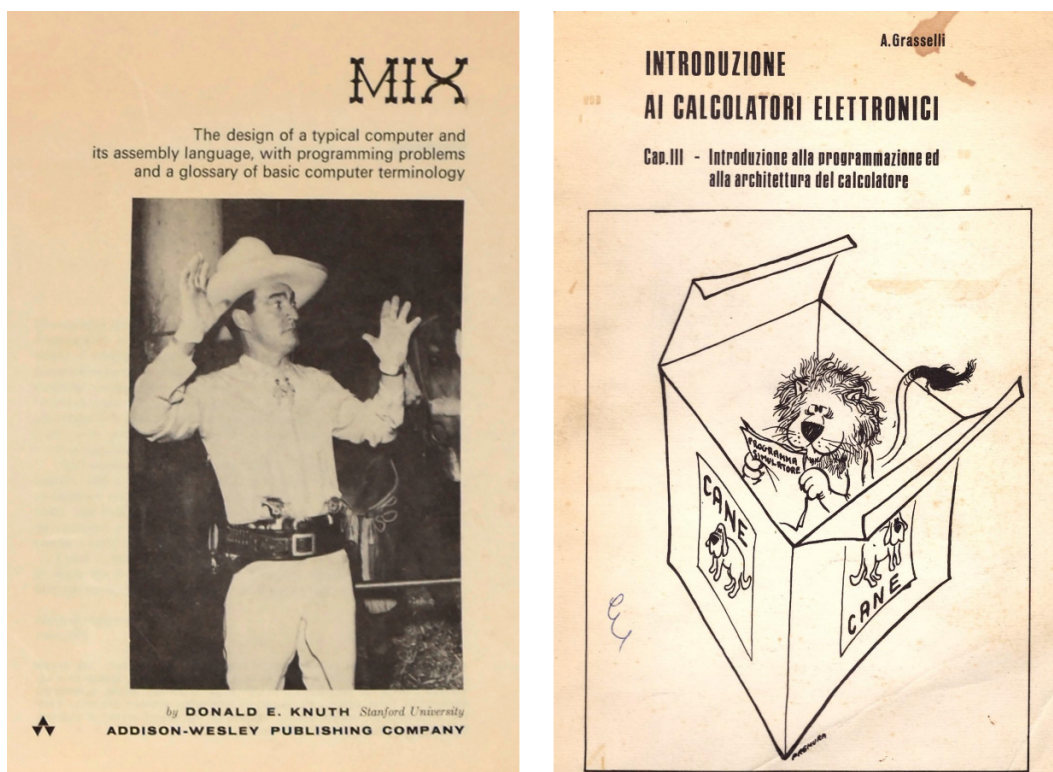


Figura 1

Copertine di calcolatori didattici a confronto, si può non essere noiosi

Non ci sono omini invece nel MIX di Knuth. Ma, a dimostrazione del divertimento genuinamente hacker con cui gli informatici affrontano la propria disciplina, anche il MIX cela parecchie curiosità. Per essere un buon esempio, il calcolatore didattico di Knuth doveva ben rappresentare le macchine reali dell'epoca. Knuth compilò perciò la lista delle macchine alle quali il MIX si ispirava e sulle quali il MIX poteva anche essere facilmente simulato. Fra i calcolatori più noti c'erano gli IBM 360, 650, 601, 709 e 7070, gli Univac SS80 e 1107, il CDC 1604, il Burroughs B220, l'Honeywell H800, il Digital PDP4... Tutte le macchine della lista di Knuth avevano un numero nel nome: un calcolatore che si rispetti deve avere delle cifre nella sigla che lo identifica, dopo tutto è una macchina digitale, cioè "a cifre". Che numero dare al calcolatore didattico? Knuth fece la media di tutti i numeri nella lista, il risultato fu 1009, che in numeri romani è MIX! Si sospetta che la lista sia

stata costruita ad arte, ma come Tom Mix, famoso attore western dei tempi del muto, di fronte a Knuth non possiamo che arrenderci (sempre fig. 1).

3. La didattica a Pisa, dalle CEP al CANE

Il CANE fu usato a Scienze dell'Informazione, nel corso di Teoria e Applicazione delle Macchine Calcolatrici (TAMC). Come strumento didattico, originale e realizzato con il contributo degli studenti, è un rappresentante ideale di quell'esperienza, anche per il ruolo che Grasselli ebbe nella progettazione del corso di laurea pisano.

Il corso di laurea fu attivato nell'a.a. 1969/70. L'iter è noto: la delibera della Facoltà di Scienze (8 marzo 1968), i passaggi al Consiglio di Amministrazione e al Senato Accademico (20 e 28 marzo), la richiesta di modifica dello statuto al Ministero (22 aprile), il parere favorevole della I Sez. del Consiglio Superiore della Pubblica Istruzione (31 ottobre) e, infine, il già citato DPR n. 24 del 28 gennaio 1969 [6] che formalmente istituì la nuova laurea. Per la storia generale rimandiamo a [12], nel seguito approfondiamo il ruolo di Grasselli, la collocazione di TAMC nel corso di laurea, l'uso del CANE e di altre macchine esempio in TAMC e in altri insegnamenti pisani.

Iniziative di formazione universitaria sui calcolatori elettronici erano attive da tempo. A Pisa dal 1964 esisteva il corso di perfezionamento in *Calcolo Automatico*. Offerte simili c'erano al Politecnico di Milano, dove lavorava il gruppo di Luigi Dadda, e a Roma, all'Istituto Nazionale per le Applicazioni del Calcolo (INAC) del CNR, ispirato, fondato e diretto da Mauro Picone. In diverse Università c'erano singoli corsi inseriti nei piani di studi di Scienze o di Ingegneria. I tempi per un corso di laurea erano quindi maturi e il progetto pisano era allineato agli orientamenti discussi sulla scena internazionale. La Association for Computing Machinery aveva promosso sin dal 1962 un comitato per la formazione in *Computer Science*; nel 1968, con il supporto della National Science Foundation, era stato pubblicato il *Curriculum 68* [13] che, definendo insegnamenti e contenuti, ispirò molti corsi di laurea in USA e in Europa.

Nel 1967 Grasselli, che veniva dal gruppo di Dadda e aveva portato a Pisa altri giovani ricercatori, aveva redatto uno studio sugli insegnamenti sui calcolatori elettronici [14], interessante anche per il corposo insieme di allegati. Il corso di laurea aveva poi preso forma e, pochi mesi prima dell'inizio delle lezioni, Grasselli lo aveva presentato a un seminario internazionale [15]. Il Curriculum 68 evidenziava le relazioni e le dipendenze con il curriculum in *Mathematics* [16]. Il corso di laurea fu attivato dalla Facoltà di Scienze e, nel programma di studi pisano, si riconoscono molti dei titoli dei corsi del Curriculum 68, propri di Computer Science, o mutuati da Mathematics. Nella presentazione di Grasselli si citano anche i promotori del corso di laurea pisano: in maggior parte docenti della facoltà di Scienze e ricercatori del CNR, ma anche rappresentanti di Ingegneria e dell'industria: IBM Italia e Italsiel.

A Pisa i primi insegnamenti con questi obiettivi furono parte della missione didattica che coinvolse i ricercatori del progetto CEP. Seminari furono organizzati già nel gennaio 1955 [17], ma diventarono presto articolati cicli di lezioni come quelle tenute da Elio Fabri nel marzo-maggio 1956 [18]. Il primo

corso ufficiale a Pisa fu di Corrado Böhm [19], in prestito dall'INAC di Picone. Oltre a testimoniare la collaborazione fra due realtà nazionali, il corso di *Calcoli Numerici e Grafici* del 1958/59 copriva temi teorici e pratici che già prefiguravano la necessità di una formazione di maggior respiro [20]. Il corso di Böhm attesta già l'idea di un calcolatore modello pensato per fini didattici: nelle lezioni illustrò una "calcolatrice semplificata" (allora "calcolatore" non indicava una macchina, ma la persona responsabile di un procedimento di calcolo).

Altra testimonianza è il corso di *Cibernetica* 1961/62 di Alfonso Caracciolo [21], protagonista della realizzazione delle due CEP. Il suo corso è in gran parte dedicato alla programmazione, per quanto riguarda l'architettura dei calcolatori, l'esempio è la seconda CEP, fresca di inaugurazione e visitata dagli studenti nell'ultima lezione.

L'insegnamento di TAMC era un complementare già presente nelle lauree in Matematica e in Fisica. A Pisa fu attivato per la prima volta nel 1967/68, inizialmente tenuto da Grasselli e Caracciolo [22, 23]. Era un classico insegnamento introduttivo, con contenuti vicini a quelli dei corsi "precursori" di Böhm e di Caracciolo: le principali basi teoriche, i principi della programmazione l'architettura degli elaboratori. Nella nuova laurea in Scienze dell'Informazione, divenne il primo contatto degli studenti con la disciplina – proprio come era *Introduction to Computing*, la radice dell'albero in cui erano organizzati gli insegnamenti del Curriculum 68.

Nel 1969/70, con l'attivazione di Scienze dell'Informazione e l'aumentato numero di studenti, ai docenti di TAMC si aggiunsero Alfio Andronico ed Eugenio Morreale: Andronico e Caracciolo coprirono i corsi per Matematica e Fisica, mentre Grasselli e Morreale si dedicarono alle edizioni di TAMC per la nuova laurea. Negli anni successivi l'insegnamento fu coperto anche da Pietro Piram e Giorgio Levi, Franco Preparata, Giuseppe Gestri, Carlo Montangero, Giuliano Pacini. Il CANE, non appena disponibile, iniziò a comparire nelle lezioni di TAMC. Esempi di come fu presentato agli studenti si trovano nei registri dei corsi di Levi/Piram (e.g. 1970/71 [24]), Montangero (e.g. 1972/73 [25]), Preparata (e.g. 1972/73 [26]), Pacini (e.g. 1974/75 [27]), Gestri (e.g. 1976/77 [28]).

Nella discussione sui metodi didattici, la dialettica fra esempi costruiti ad hoc ed esempi tratti dal mondo reale è presente e vivace in molte discipline. Per l'informatica, il CANE, come LMC e MIX, è un rappresentante degli esempi ad hoc: intorno alla sua storia si trovano le tracce di quella dialettica nel corso di laurea pisano. Per esempio, già nel 1972/73, al termine del suo corso Preparata presentava anche il *Digital PDP-8*. Fu però un altro Digital storico, il *PDP-11*, che con continuità prima affiancò e poi sostituì il CANE per illustrare agli studenti architetture reali e più moderne. I corsi di Montangero rappresentano bene la transizione fra il 1977/78 [29] e il 1980/81 [30]. Il *MiniCANE*, protagonista delle citate lezioni televisive, testimonia un'evoluzione: era scritto in *Algol-W* (un derivato dell'*ALGOL 60*, proposto da Niklaus Wirth e precursore del *Pascal*) e fu usato nei corsi (e.g. Montangero 1979/80 [31]). Il più fedele al CANE fu Gestri: lo usò nelle sue lezioni fino al 1990/91 [32].

Oltre all'uso didattico è interessante anche l'origine didattica del CANE. Il simulatore del CANE fu oggetto della tesi di Pacini discussa nel marzo 1970. Un altro tesista, Attilio Ripoli, nel luglio 1970 si laureò lavorando sul cross-assemblatore: girava sul 7090 per produrre codice per il CANE. Relatore delle due tesi fu Grasselli ed entrambe furono lauree in Fisica. Si potrebbe dire che il CANE segna il passaggio di testimone dai fisici, che avevano dato il via alla storia con il progetto CEP, agli informatici, che iniziavano a esistere ufficialmente come "prodotti" del nuovo corso di laurea.

4. Il CANE, architettura e uso

Il CANE era pensato per avvicinare gentilmente gli studenti alla comprensione intima del funzionamento dei calcolatori: una consapevolezza che non si acquisisce programmando solo in linguaggi ad alto livello. Il concetto è ben spiegato da Grasselli nelle prime pagine della sua dispensa [10] con una metafora che invitiamo a leggere direttamente. Grasselli avverte, citando Konrad Lorenz, che la cosa non è senza impegno.

Perciò, per aiutare il cammino degli studenti, il CANE era nato semplice [33]. La memoria era di 512 celle di 18 bit, sufficienti per gli esercizi di programmazione. Il processore aveva 2 registri di lavoro a 18 bit e 7 registri indice a 9 bit usabili come contatori o modificatori d'indirizzo. Per comprendere la macchina, agli studenti erano esposti anche il contatore di programma, il registro dell'istruzione corrente, i registri di indirizzamento e di lettura/scrittura della memoria.

Le istruzioni macchina erano 64. Trasferimenti fra memoria e registri, operazioni aritmetiche e logiche, confronti e salti ben rappresentavano le istruzioni tipiche delle macchine del tempo. Facevano invece eccezione le istruzioni di utilità per le conversioni binario/decimale e per la gestione del lettore di schede e della stampante. Le prime, sofisticate e complesse, nei calcolatori reali sono di solito realizzate come routine di sistema e non come istruzioni di macchina. Le seconde sono proposte nel CANE in una forma semplificata che nasconde molti dettagli della gestione delle periferiche. Entrambi i casi, con soluzioni diverse, sono espressioni della natura didattica del CANE: affinché gli studenti si concentrassero sugli aspetti più interessanti degli algoritmi assegnati come esercizi di programmazione, erano loro risparmiati compiti noiosi (le conversioni) o troppo difficili (la gestione delle periferiche).

Il CANE aveva una consolle. Mostrava lo stato di tutti i registri, dei bit di confronto e di traboccamento; ospitava i pulsanti per la sequenza di avvio da lettore di schede e per il controllo dell'avanzamento singolo o continuo dei programmi. La consolle però non era simulata: d'altra parte, gli studenti non avrebbero potuto vederla nel contesto operativo del CNUCE che ospitava l'IBM 7090 sul quale girava il SimulCANE.

Il 7090 era stato inaugurato nel 1965 insieme all'apertura del Centro. Proprio nei primi anni di Scienze dell'Informazione arrivarono al CNUCE macchine più moderne: l'IBM 360/67 nel 1971 [34] e l'IBM 370/155 nel 1973 [35]. Il CANE fu aggiornato per girare sul 360/67: nell'estate del 1972 il simulatore fu riscritto in PL/I

da Francesco Romani, ancora uno studente, questa volta però di Scienze dell'Informazione. Ma il 7090 fu mantenuto in servizio per diversi anni ancora [36].

In ogni caso, gli studenti avevano accesso al CANE attraverso le procedure con cui tutti gli utenti usavano i calcolatori del CNUCE [37]: in orari prestabiliti i programmi (pacchi di schede perforate) erano consegnati agli operatori del centro, le esecuzioni erano pianificate per la giornata successiva, alla fine della quale i risultati (stampe su foglio continuo) erano disponibili per essere ritirati.

Quindi, la consolle del CANE era descritta solo per dare agli studenti un'idea concreta della macchina e permettere loro di figurarsi le azioni di un operatore che caricava i programmi del CANE e ne controllava l'esecuzione.

Per eseguire i programmi scritti per il CANE, il pacco di schede doveva iniziare con i due programmi per il 7090 *SystemCANE* e *SimulCANE*. Seguivano le schede dei programmi nel linguaggio del CANE. Caricati e messi in esecuzione sul 7090 i primi due, le schede successive erano interpretate dal CANE simulato.

Il simulatore web realizzato oggi permette invece di lavorare con il CANE come un operatore: esiste una consolle e si ha accesso a due tipiche periferiche del 7090, il lettore di schede *IBM 1402* e la stampante *IBM 1403*, come se fossero collegate al CANE. Sul lettore si può caricare direttamente un programma per il CANE.

Come tutte le ricostruzioni, anche quella del CANE ha posto dilemmi e richiesto scelte. Sono stati risolti cercando di interpretare al meglio le intenzioni didattiche originali. Per esempio, già l'accesso alla consolle e alle periferiche è una deviazione rispetto all'originale. È motivata dalla volontà di offrire oggi quel che le tecnologie di ieri non permettevano, ma che gli autori avrebbero desiderato pensando allo scopo del CANE: spiegare il funzionamento dei calcolatori. Altre scelte sono decisamente più tecniche, ne discutiamo brevemente un paio.

Secondo [4, 10] il CANE rappresenta i numeri negativi in complemento a 2. Tuttavia, negli stessi documenti, si dichiara un intervallo simmetrico, tipico della rappresentazione in modulo e segno. La contraddizione è rimasta insoluta: per non far pensare a un errore il CANE web funziona con la rappresentazione in complemento a 2 standard e gli interi a 18 bit vanno da -217 a $217-1$.

La divisione era stata implementata con un algoritmo descritto in [38]. È una divisione particolare per la quale, per esempio, $128 \div 10$ fa 13 con resto -2 . Nel CANE web si è scelto la divisione standard e il resto ha sempre il segno del dividendo.

In generale, il criterio adottato nel risolvere contraddizioni e "stranezze" è stato di facilitare l'uso a chi si avvicina al CANE per capire come funziona un calcolatore. Soprattutto considerando che il CANE web ha anche il senso di mostrare agli utenti dell'informatica di oggi com'era fatto e come si usavano i calcolatori mezzo secolo fa.

5. Ciao Mondo sul CANE

Per mettersi nei panni degli studenti di TAMC degli anni '70 sarebbe necessario un po' di studio, partendo dalla dispensa di Grasselli [10] per poi approfondire con la nota tecnica [33] e le tesi [4, 5]. Però, per soddisfare l'impazienza di vedere il CANE in funzione, non poteva mancare il più classico degli esempi di programmazione.

Storicamente, occorre segnalare che gli studenti che usavano il CANE non potevano conoscere *Hello World!* Comparve per la prima volta nel 1974 in un memorandum di Brian Kernighan rimasto interno ai *Bell Labs*. Fu poi ripreso in "The C Programming Language", scritto con Dennis Ritchie e pubblicato nel 1978 [39]. Negli anni a seguire, il libro divenne una pietra miliare, oltre che per il C, anche come modello di libro sulla programmazione: dalla formattazione tipografica, allo stile della scrittura, al programmino di primo contatto. Sul CANE inoltre è CIAO MONDO, le minuscole e il punto esclamativo non c'erano nell'insieme dei caratteri che il CANE ereditava dalle periferiche dell'IBM 7090.

Un programma per il CANE, nella sua forma più elementare, fisicamente consiste in un pacco di schede perforate in binario (fig. 2), ognuna contenente 4 parole di 18 bit. La prima scheda è per il *caricatore minimo*, il programmino (4 istruzioni, una scheda) che carica nella memoria del CANE le schede successive, cioè il vero programma utente, anch'esso codificato in binario. La versione minimale di CIAO MONDO si programma in 7 parole, 3 di istruzioni e 4 di dati (la stringa da stampare). In tutto sono 3 schede: 1 di caricatore minimo e 2 di programma.

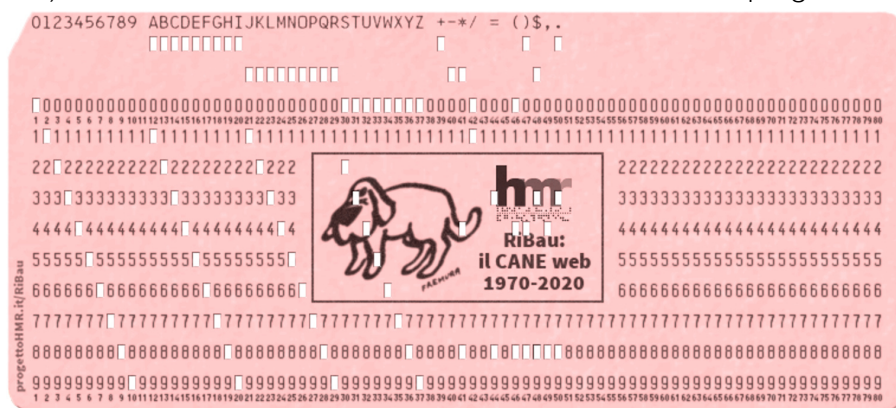


Figura 2

Una scheda perforata, sulle colonne i fori codificano i caratteri stampati in alto

Il CANE web legge le schede come sequenze di caratteri. Le schede infatti questo erano: le perforazioni su una colonna codificavano un carattere; il perforatore, per facilitare la lettura, stampava in alto il carattere corrispondente. Le cifre '0', '1'... '9' erano codificate con un solo foro nella riga corrispondente al valore numerico, quindi le schede binarie avevano perforazioni solo sulle prime due righe. In pratica, le 3 schede di CIAO MONDO sono:

```
0001110011111100011101000100000110001010000100000010010000000000000001
11110100000000010011110000000000011100000000000000101010011011001010001
1001101100001001001001101001010100100110110000110000
```

Ogni riga corrisponde a una scheda, la terza non è usata completamente. Per caricare le schede sul CANE web (vedi fig. 3) è sufficiente premere il pulsante *Load Cards* sul lettore di schede IBM 1402 e “caricare” le schede copiando il testo nella dialog box che si apre. Le schede appaiono sul carrello e lo stato del lettore diventa *Ready*. Cliccando sulle schede si può vedere l’aspetto reale di una scheda. Per eseguire CIAO MONDO basta premere il pulsante *L* sulla consolle del CANE.

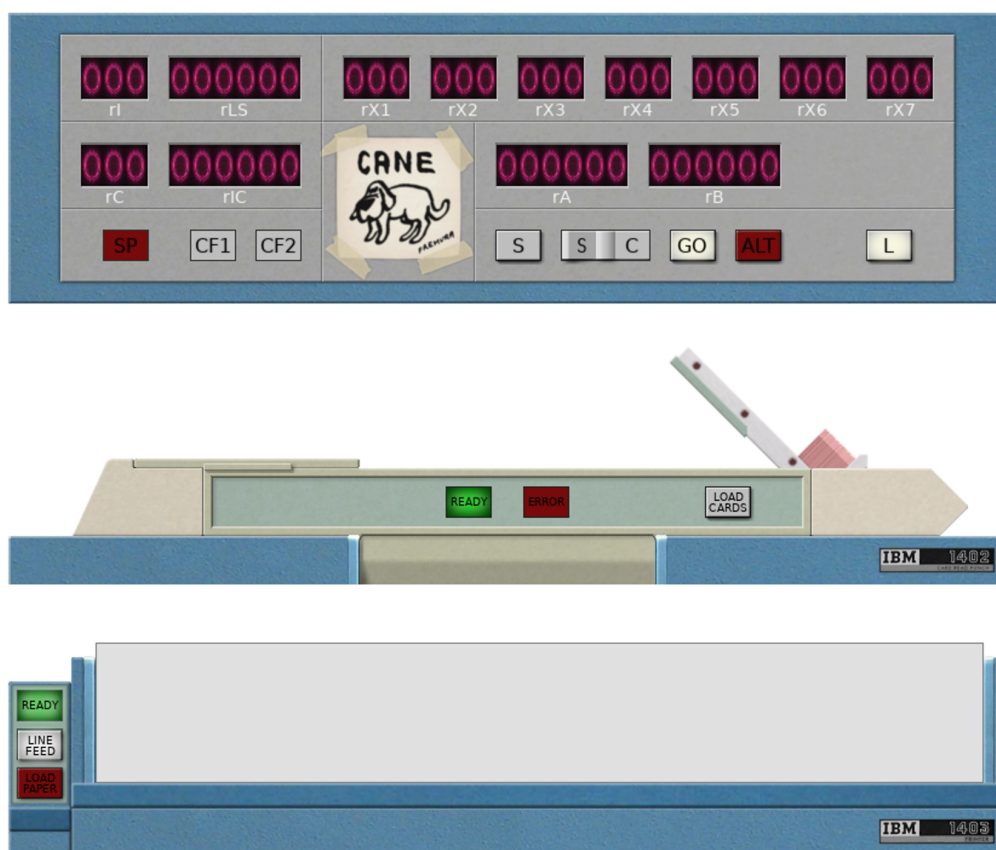


Figura 3

Il CANE come si presenta sul web: la consolle, il lettore di schede, la stampante

Di seguito è descritto cosa succede in dettaglio nel CANE.

La pressione di *L* fa partire la sequenza di avvio da lettore di schede. Questa legge una scheda e trasferisce le 4 parole in memoria a partire dalla locazione 000. Terminata la lettura il CANE comincia ad eseguire le istruzioni a partire dalla locazione 000. Quindi il risultato della pressione di *L* è caricare e lanciare il caricatore minimo.

Il caricatore minimo legge le schede che lo seguono, 2 nel nostro caso, ne trasferisce il contenuto in memoria a partire dalla locazione 004 e, terminata la lettura, esegue un salto a 004. Il risultato è caricare e lanciare il vero CIAO MONDO.

Il programma CIAO MONDO è fatto di 3 istruzioni codificate nei primi 18+18+18 bit della seconda scheda. Segue la stringa da stampare: "CIAO MONDO " con due spazi in fondo per fare 12 caratteri che, nella codifica IBM a 6 bit per carattere, a 3 caratteri per parola, costituiscono le ultime 4 parole del programma.

Scrivere ed eseguire programmi in binario era una via sicura per acquisire una conoscenza profonda dell'organizzazione e del funzionamento dei calcolatori. Tuttavia, la fatica di scrivere e leggere i bit comprometteva la capacità di ragionare sulla logica degli algoritmi. Nella progressione didattica di cui il CANE era strumento, erano previsti altri passaggi.

Un passo intermedio era la codifica ottale: pur essendo solo una riscrittura compatta del binario era già più leggibile per lo studente.

Il livello più alto era un linguaggio simbolico in stile Assembler, il *SimbolCANE*, derivato da una semplificazione e adattamento del *MAP* del 7090 [40]. Con il *SimbolCANE* lo studente poteva esercitarsi a scrivere algoritmi di una certa complessità senza confondersi con le codifiche numeriche, ma anche senza perdere di vista l'architettura del calcolatore dato che a ogni riga simbolica corrispondeva esattamente un'istruzione macchina.

Il binario di CIAO MONDO perforato sulla seconda e terza scheda è generato dal programma assembler [5] a partire dal codice in fig. 4, dove a fianco delle righe in *SimbolCANE* si mostrano i corrispondenti codici ottali e binari.

SIMBOLCANE	MEM.	OTTALE	BINARIO
ORIG 4	...		
IN CEN IN	004	750004	111 101 000 000 000 100
USC CM	005	740007	111 100 000 000 000 111
ALT IN+1	006	000005	000 000 000 000 000 101
CM ALFA 4, CIAO MONDO	007	233121	010 011 011 001 010 001
FINE IN	010	466044	100 110 110 000 100 100
	011	464524	100 110 100 101 010 100
	012	466060	100 110 110 000 110 000
	...		

Figura 4
CIAO MONDO in SimbolCANE

Come prescritto dalla pseudo-istruzione ORIG, le 3 istruzioni eseguibili e le successive 4 parole di dati alfanumerici vanno caricate in memoria a partire dalla locazione 004 in poi. Sia i codici operativi di macchina (CEN, USC e ALT) che i riferimenti alle celle della memoria (IN e CM) sono espressi in modo simbolico. Nella pseudo-istruzione FINE si dice che la prima istruzione eseguibile di CIAO MONDO si trova alla locazione simbolica IN, cioè la cella 004.

La prima istruzione del programma è una CEN, per “controllo entrata”: serve solo ad assicurarsi che il lettore di schede abbia terminato di scrivere in memoria.

La seconda istruzione è una USC per “uscita”: dice alla stampante di stampare una riga, cioè 120 caratteri, presi 3 a 3 da 40 parole consecutive della memoria a partire da CM, cioè 007 da dove inizia la stringa. Sulla stampante, dopo “CIAO MONDO ”, la riga sarà completata con zeri, cioè con il contenuto di default della memoria che corrisponde alla codifica del carattere ‘0’.

La terza istruzione è una ALT, che arresta l’esecuzione saltando a IN+1, ovvero 005. In questo modo il CANE è fermo ma posizionato sulla USC, cioè pronto per ristampare “CIAO MONDO ”. Basta premere il pulsante GO sulla console.

6. Conclusione

L’idea di far rivivere il CANE, oltre che dalla curiosità propria della ricerca storica e dal divertimento della ricostruzione, è nata per portare ai festeggiamenti del 50° di Scienze dell’Informazione un contributo che fosse particolarmente rappresentativo di quell’esperienza didattica.

È però motivata anche dalla convinzione che i calcolatori didattici siano tuttora utili a comprendere l’informatica: a maturare la rivelazione del “ora so davvero come funziona”, cioè quella comprensione intima che Grasselli voleva trasmettere ai suoi studenti.

I dispositivi dell’informatica di oggi, sotto la superficie amichevole, nascondono sistemi incredibilmente complessi, difficili se non impossibili da comprendere appieno. Il CANE, nella sua antica semplicità restituisce allo studente la soddisfazione di capire veramente cosa sia l’informatica. Dietro a RiBau c’è l’augurio che il CANE possa ancora essere utile.

Ringraziamenti

Grazie a Linda Pagli per aver reso disponibile la dispensa con i disegni di Fremura e a Carlo Montangero che ha scovato chi ne aveva conservato una copia. L’Archivio Generale di Ateneo e la Direzione Servizi per la Didattica hanno permesso l’accesso e la riproduzione dei documenti, un grazie a Luigi Rivetti per l’interessamento. Grazie ad Andrea Pachetti che ci ha segnalato il video con le lezioni di Grasselli sulla RAI.

Riferimenti

Insieme al simulatore, la documentazione originale del CANE e gran parte del materiale d’archivio citato è disponibile in forma digitale sulle pagine web di Progetto HMR.

[1] <http://progettoHMR.it> (ultimo accesso novembre 2020).

[2] Cignoni, G.A., Ceccarelli D., Imbrenda C. (2009). “Il ‘restauro’ del software di sistema della Macchina Ridotta del 1956”, *Atti del 47mo Congresso Nazionale AICA, Roma*, 1-11.

[3] Cignoni, G.A., Gadducci F., Paci S. (2015). “A Virtual Experience on the Very First Italian Computer”, *ACM Journal on Computing and Cultural Heritage*, 7(4), 1-23.

- [4] Pacini, G. (1970). *Architettura e simulazione di un calcolatore didattico* (tesi di laurea in Fisica, Università di Pisa, rel. Grasselli A.).
- [5] Ripoli, A. (1970) *Un programma assemblatore per un calcolatore didattico* (tesi di laurea in Fisica, Università di Pisa, rel. Grasselli A.).
- [6] "DPR 24 del 18 gennaio 1969" in *Gazzetta Ufficiale della Repubblica Italiana*, 64 (1969).
- [7] Yurcik, W., Osborne H. (2001). "A Crowd of Little Man Computers: Visual Computer Simulator Teaching Tools", *Proceedings of the 33rd conference on Winter Simulation Conference (WSC 2001)*, Arlington VA, 1632-1639.
- [8] Knuth, D. (1968). *The Art of Computer Programming. Vol. 1, Fundamental Algorithms*, Addison-Wesley.
- [9] Englander, I. (2014⁵). *The Architecture of Computer Hardware, Systems Software, and Networking: An Information Technology Approach*, John Wiley & Sons.
- [10] Grasselli, A. (1972). *Introduzione ai calcolatori elettronici. Cap. III, Introduzione alla programmazione e all'architettura del calcolatore*, Università di Pisa.
- [11] Grasselli, A. "Le istruzioni del MiniCane", <http://youtube.com/watch?v=A24eAGoZAuc> (ultimo accesso novembre 2020).
- [12] Montangero, C. (2013). "I corsi di laurea di Scienze (in Informatica)", *Quaderni della Fondazione Galileo Galilei*, 1.
- [13] ACM Curriculum Committee on Computer Science (1968). *Curriculum 68. Communications of the ACM*, 11(3) 151-197.
- [14] Grasselli, A. (1967). "Insegnamento universitario nel campo dei calcolatori elettronici", *IEI Nota interna*, II.61.
- [15] Grasselli, A. (1969). "Computer Science at The University of Pisa", *Development and Organization Criteria for University Curricula in Computer Science, Varese. IEI Nota interna*, B69/16.
- [16] MAA Committee on the Undergraduate Program in Mathematics (1965). *A general curriculum in mathematics for colleges*, Mathematical Association of America.
- [17] *Riassunto della riunione all'Istituto di Fisica del 13-14 gennaio 1955*, Università di Pisa.
- [18] Fabri, E. (1956). "Appunti dalle lezioni di Introduzione alla Programmazione di una Calcolatrice Elettronica", *Nota interna CSCE*, 1/35.
- [19] Böhm, C. (1959). *Registro delle lezioni di "Calcoli numerici e grafici" dell'a.a. 1958/59*, Università di Pisa.
- [20] Cignoni, G.A., Gadducci, F. (2013). "A Syllabus for the Fifties, Teaching Computer Science on the first Italian Computers", *2nd International Conference on the History and Philosophy of Computing*, Paris.
- [21] Caracciolo, A. (1962). *Registro delle lezioni di "Cibernetica" dell'a.a. 1961/62*, Università di Pisa.

- [22] Università di Pisa (1968). *Annuario dell'Università degli studi di Pisa per l'a.a. 1967/68*, Università di Pisa.
- [23] Università di Pisa (1969). *Annuario dell'Università degli studi di Pisa per l'a.a. 1968/69*, Università di Pisa.
- [24] Levi, G., Piram, P. (1971). *Registro delle lezioni di "TAMC" dell'a.a. 1970/71*, Università di Pisa.
- [25] Montangero, C. (1973). *Registro delle lezioni di "TAMC" dell'a.a. 1972/73*, Università di Pisa.
- [26] Preparata, F. (1973). *Registro delle lezioni di "TAMC" dell'a.a. 1972/73*, Università di Pisa.
- [27] Pacini, G. (1975). *Registro delle lezioni di "TAMC" dell'a.a. 1974/75*, Università di Pisa.
- [28] Gestri, G. (1977). *Registro delle lezioni di "TAMC" dell'a.a. 1976/77*, Università di Pisa.
- [29] Montangero, C. (1978). *Registro delle lezioni di "TAMC" dell'a.a. 1977/78*, Università di Pisa.
- [30] Montangero, C. (1981). *Registro delle lezioni di "TAMC" dell'a.a. 1980/81*, Università di Pisa.
- [31] Montangero, C. (1980). *Registro delle lezioni di "TAMC" dell'a.a. 1979/80*, Università di Pisa.
- [32] Gestri, G. (1991). *Registro delle lezioni di "TAMC" dell'a.a. 1990/91*, Università di Pisa.
- [33] Grasselli, A., Pacini, G. (1970). "CANE SimulCANE e SystemCANE", *IEI nota tecnica C70/1*.
- [34] Marconi, R. (1971). *Inizio del servizio del Sistema IBM 360/67* (CNUCE, Comunicazione del Direttore, 6 agosto), Università di Pisa.
- [35] Torrigiani, G. (1973). *Utenza CNUCE Sistema IBM 370/155* (CNUCE, Comunicazione del Comitato Direttivo, 20 marzo), Università di Pisa.
- [36] Bronzoni, P. (1973). *Servizio 7090* (CNUCE, Comunicazione del responsabile del Servizio Elaborazione Dati, 20 marzo), Università di Pisa.
- [37] Faedo, A. (1967). *Norme relative all'utenza del Sistema 7090* (CNUCE. Comunicazione del Direttore, 23 marzo), Università di Pisa.
- [38] Chu, Y. (1962). *Digital computer design fundamentals*, McGraw-Hill.
- [39] Kernighan, B.W., Ritchie, D. (1978). *The C Programming Language*, Prentice Hall.
- [40] IBM. (1966). *IBM 7090 IBSYS Operating System. Macro Assembly Program (MAP) Language*, IBM Systems Reference Library.

Biografie

Alessandro Cignoni (Progetto HMR)

Studente triennale di Informatica Umanistica presso l'Università di Pisa, prossimo alla laurea, è anche collaboratore del Progetto HMR. Fra i suoi interessi spiccano la storia e le tecnologie Geographic Information System (GIS). In RiBau si è occupato principalmente della programmazione in Javascript dell'interfaccia del simulatore del CANE.

Email: alessandro.cignoni@progettoHMR.it

Giovanni A. Cignoni (Progetto HMR)

Laureato in Scienze dell'Informazione e poi ingegnere, lavora nel trasferimento tecnologico occupandosi di ingegneria del software in più declinazioni. Ha pubblicato, partecipato a progetti internazionali, insegnato alle università di Pisa, Firenze e Padova. Dal 2006 si dedica anche a HMR, un progetto di ricerca che, con l'ostinata curiosità degli hacker, studia la storia dell'informatica a partire da hardware e software. Dal 2015 è titolare di Storia dell'Informatica alla laurea in Informatica Umanistica, Università di Pisa.

Email: giovanni.cignoni@progettoHMR.it

Giuliano Pacini (Progetto HMR)

Il Prof. Pacini si laurea in Fisica a Pisa nel 1970. Nel 1973 è Assistente Ordinario presso Scienze dell'Informazione di Pisa, dove diventa Prof. Associato. Dal 1987 insegna come Prof. Ordinario all'Università di Salerno. Nel 1991 si trasferisce a Venezia e infine all'A.N. di Livorno. Va in pensione per limiti di età nel 2012. Ha fatto ricerca pubblicando in diversi settori sia teorici che applicativi dell'Informatica: Linguaggi di programmazione, Software Engineering, Logic Programming, Linguaggi Visuali, Sistemi per la Didattica.

Email: giuliano.pacini@progettoHMR.it

Daniele Ronco (Archivio di Ateneo, Università di Pisa)

Da sempre si occupa di libri e di biblioteche e da una ventina d'anni cura l'Archivio dell'Università di Pisa.

Email: daniele.ronco@sba.unipi.it