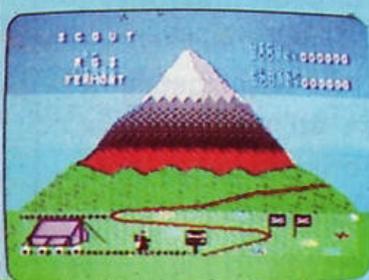


# GO GAMES

mensile d'informatica e video-games - 14 - novembre '86 - L. 8.000

**7 VIDEO-GAMES**  
per  
**CBM 64 e 128**

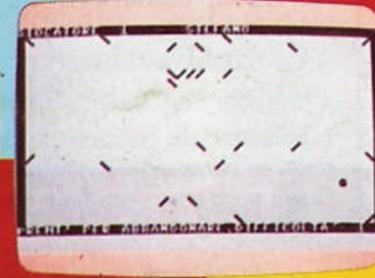
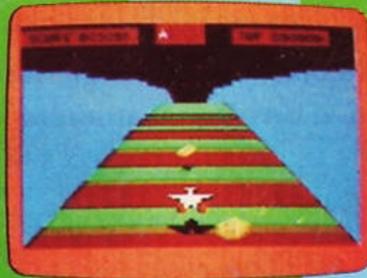
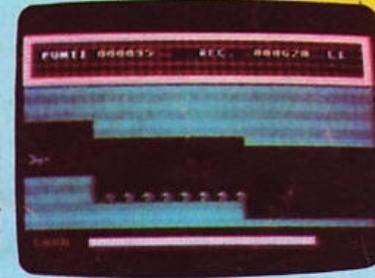
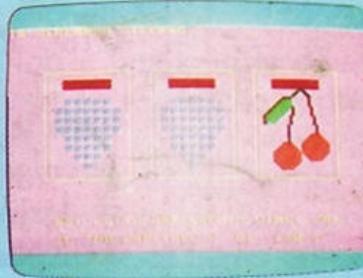
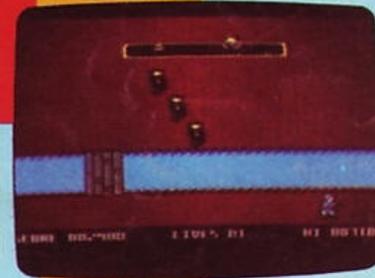
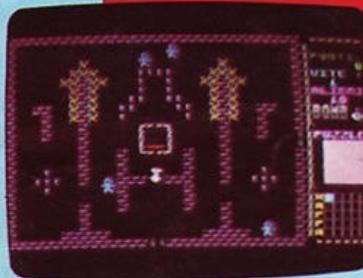
**7 VIDEO-GAMES**  
per  
**C 16 e PLUS 4**



- 1 - FUNGUS
- 2 - HELIRAID
- 3 - SUPER TUONO
- 4 - SCOUT
- 5 - HELIKOPTER
- 6 - SPACE BLITZ
- 7 - QUASAR



- 1 - RAIDERS
- 2 - TIME BOMBS
- 3 - SKRAMBLER
- 4 - RANCH
- 5 - INCURSORE
- 6 - BOUNDER
- 7 - SLOT MACHINE



# GO GAMES

Mensile di informatica  
e video giochi

Anno II  
N. 14 - Novembre '86

EDITORE:  
Editions Fermont s.r.l.  
20121 Milano

REDAZIONE:  
Via Cialdini, 11  
20161 Milano  
Tel. 02/6453775/6

FOTOLITO:  
Claudio Lavezzi  
Via Terruggia, 3  
20162 Milano

STAMPA:  
A.G.E.L. s.r.l.  
Viale dei Kennedy, 92  
20027 Rescaldina

DISTRIBUZIONE:  
MePe  
V.le Famagosta, 75  
20142 Milano

DIRETTORE RESPONSABILE:  
Amilcare Medici

Fotografie di Stefano Monti

**Numeri arretrati:** Ogni numero arretrato £. 8.000 più £. 3.000 di spese postali - Versamento da effettuare sul c/c postale n. 37332202 intestato a EDITIONS FERMONT, Via Cialdini, 11 20161 Milano

**ATTENZIONE:**  
I bollettini dei versamenti devono essere scritti in stampatello o a macchina. Quelli poco chiari verranno cestinati.

## ATTENZIONE

### CBM 64

Per il CBM 64 ti proponiamo un nuovo sistema di caricamento che ti permette di scegliere il gioco che vuoi caricare e di posizionare il nastro con l'avanzamento veloce (F.FWD) subito prima del gioco da te prescelto, quindi di procedere al caricamento normale. Con questo sistema eviti di dover passare tutto il nastro per cercare il programma che ti interessa.

Le operazioni da fare sono:

- 1) Digita Load e premi Return.
- 2) Attendi che sul video compaia la presentazione.
- 3) Premi Stop sul registratore.
- 4) Dopo qualche secondo apparirà una schermata con l'elenco dei giochi preceduto da un numero e la scritta «Programma N°» col cursore che lampeggia.
- 5) Inserisci il N° corrispondente al programma desiderato e premi «Return».
- 6) Comparirà la scritta «premi F.FWD» quindi il registratore si fermerà subito prima del programma da te scelto. A questo punto premi «STOP» e successivamente premi «PLAY».

**AVVERTIMENTO:** se lo schermo si riempirà di righe colorate significa che il caricamento procede regolarmente. Se non escono le righe torna indietro all'inizio del gioco e premi nuovamente Play.

### C16 / PLUS 4

Ecco le istruzioni per il caricamento dei programmi: Avvolgere completamente la cassetta dalla parte che si desidera caricare. Quindi digitare LOAD & RETURN e far iniziare il caricamento. Quando ricompare il cursore digitare RUN & RETURN ed attendere. La prima volta che si caricano i programmi conviene azzerare il contatore del registratore alla fine dell'avvolgimento e scrivere il numero dell'inizio del gioco in modo che in un tempo successivo si conosce l'esatto inizio del gioco.

# l'uomo e il computer 12

Certamente la storia del computer, come ogni altra storia, non segue un'unica linea: ad ogni stadio, ad ogni punto di arrivo si diparte una ramificazione, si aprono nuovi orizzonti e campi di interesse per cui per seguirne gli sviluppi è indispensabile procedere zigzagando e saltando di palo in frasca tra i vari settori.

La scorsa puntata ci siamo incontrati con Babbage e la prima macchina analitica: la prima idea di programmazione e di magazzino di dati. Nello stesso periodo però "crescevano e si moltiplicavano" anche i vari tentativi di rendere sempre più efficienti e veloci le normali macchine da calcolo. Quelle che oggi hanno le dimensioni di un biglietto del tram e, consumando sempre meno energia, la riescono a succhiare dal sole, dalla pulsazione arteriosa e forse, in un futuro, molto prossimo, dal semplice calduccio dell'essere tenute in tasca.

Ma negli ultimi anni del secolo scorso sarebbe stato estremamente complesso e voluminoso l'abbinamento tra un delicato meccanismo ad orologeria (tali erano allora le prime calcolatrici) e la corrente elettrica di una ingombrante e rudimentale pila. Il funzionamento di queste macchine era quindi meccanico, inteso come rotazione di manovelle, movimenti di leve e pignoni in scatole che potevano essere metalliche e finemente ceselate (ricordate le immagini di qualche mese fa?) oppure in rozzi contenitori di legno come quello che vediamo in fig. 1.

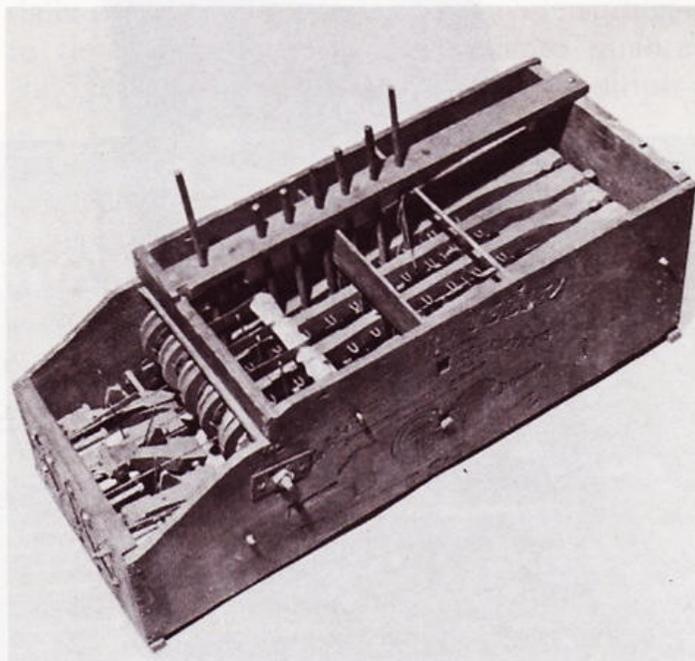


Fig. 1

Non è una scatola da scarpe, bensì una cassetta in legno che conteneva all'origine della pasta. E con il nome, appunto, di "macaroni box" fu costruita nel 1884 da un giovane meccanico di Chicago che servendosi di asticcioline di legno ed elastici ne derivò il primo prototipo della Comptometer che brevettò nel 1887.

L'ingegnoso meccanico si chiamava Dorr E. Felt ed il suo primo modello di calcolatore era basato su dei movimenti molto semplici: "I tasti disposti in colonna permettono di esercitare delle pressioni in diversi punti di

una leva portante alla sua estremità un settore dentato ingranante con una ruota solidale al numeratore. Secondo il punto ove questa leva è premuta, per l'abbassamento di un tasto, la corsa del settore è di 1, 2, 3, ...9 denti. Abbandonato il tasto, il settore, nel tornare alla posizione primitiva, ingrana in una ruota che fa avanzare il numeratore relativo del numero di unità volute". (1)

Il motto di Felt era "I contabili e gli impiegati non possono essere serviti meglio che da macchine progettate per contabili e impiegati" (2). Per questo la comptometer (vedi fig. 2).



Fig. 2

venne usata soprattutto negli uffici anche se, nel 1895, la Marina degli Stati Uniti se ne servì per tutti i calcoli richiesti dalla progettazione navale.

Macchine come questa e la successiva di William Seward Burroughs sono delle semplici addizionatrici, cioè per procedere ad una moltiplicazione si deve ricorrere a successive addizioni e similmente, per le divisioni, alle sottrazioni.

Quella di Burroughs merita una piccola digressione aneddotica. Figlio di un meccanico,

a 24 anni rilevò l'azienda del padre e la trasformò per creare l'American Arithmetometer Cy. L'utilizzazione della sua prima macchina calcolatrice era così complessa che, si dice, il solo Burroughs era in grado di usarla. Quindi, in una crisi di scoraggiamento, prese i cinquanta esemplari già costruiti e li gettò dalla finestra. Ma in seguito, grazie a nuovi dispositivi ed accorgimenti tecnici, l'American Arithmetometer Cy divenne uno degli "utensili" fondamentali che cominciarono a diffondersi negli uffici di tutto il mondo come efficace strumento per sveltire la contabilità.

Queste innovazioni consistevano principalmente nella possibilità di una "hardcopy" cioè l'uscita dei risultati, anche quelli parziali, stampati su carta.

Inoltre le "tastiere complete", cioè formate da diverse colonne di numeri (una per ogni posizione della cifra nel numero), furono sostituite dalle "tastiere ridotte" in cui la successione temporale delle cifre premute dà luogo al numero richiesto.

In fig. 3 è riprodotta una delle prime "addizionatrici portatili" ed a tastiera ridotta; l'Eureka, progettata e costruita in Svizzera nel 1907.

Nonostante il suo disegno elegante è difficile vederla come oggetto tascabile ed alla portata di tutti magari come sussidio in un difficile compito di matematica...

Continua nel frattempo la ricerca dei metodi e dei mezzi per realizzare automaticamente anche la moltiplicazione. Nel 1883 fu prodotto il "moltiplicatore Eggis", un incrocio tra il regolo calcolatore e la vera e propria calcolatrice.

Per operare in forma diretta con la moltiplicazione è necessario "memorizzare" o comunque servirsi della tavola pitagorica. Nove di queste (in ognuna delle quali figurano i prodotti delle cifre da 1 a 9) sono riportate su dei cartoncini sovrapposti che, per lo spostamento di nove guide, danno luogo ai risultati parziali della moltiplicazione. Una lavagna, compresa nel prezzo e nel rudimentale hardware, permette poi di sommarli ed ottenere il risultato finale. (vedi fig. 4)

Ma dobbiamo aspettare il 1887 e il francese

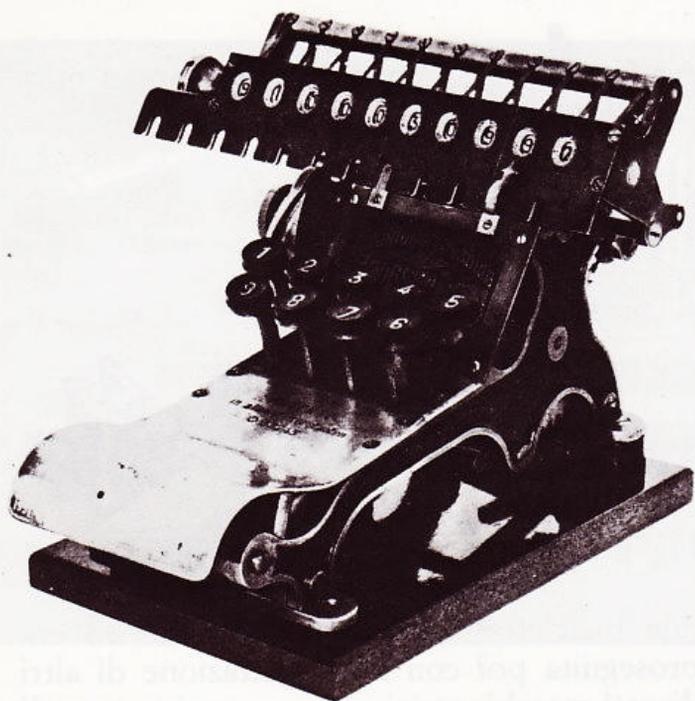


Fig. 3

Léon Bollée che a soli diciott'anni costruisce la prima calcolatrice che esegue automaticamente anche le moltiplicazioni ma, ahimè, non le divisioni.

“Con la macchina di Bollée si attua il calcolo con un'economia dell'80% del tempo necessario con le altre calcolatrici. (...) Una tavoletta che costituisce la rappresentazione materiale della tavola pitagorica, poiché comprende nove linee e nove colonne di asticcioline le cui lunghezze rappresentano, rispettivamente, la cifra delle decine e quella delle unità del prodotto che nella tavola pitagorica occupa la medesima posizione. (...) I numeratori ricevono il loro movimento da aste dentate i cui spostamenti sono comandati dalle coppie di asticcioline delle tavole suddette (le quali dovranno essere tante quante sono le cifre del moltiplicando). Per moltipli-

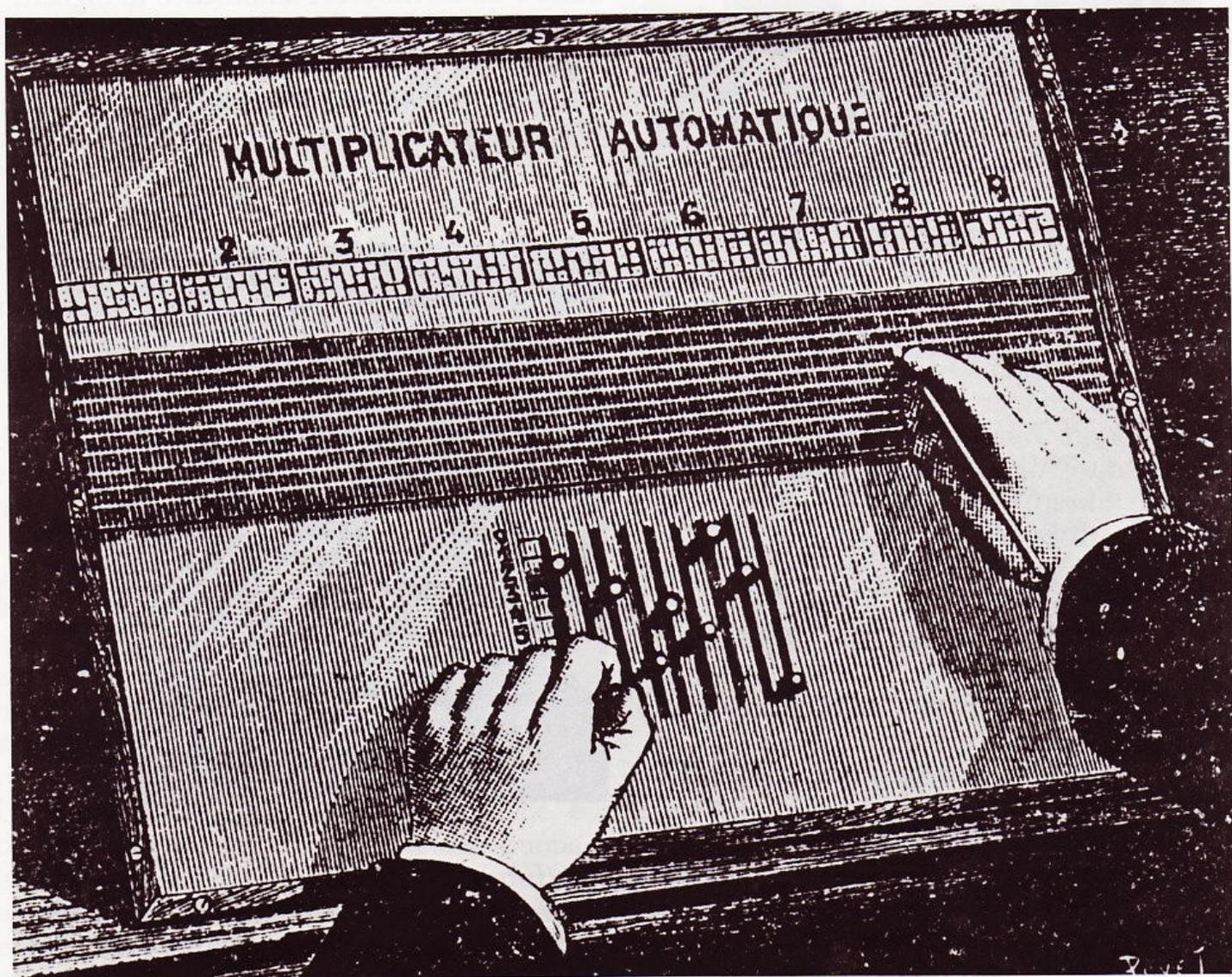


Fig. 4

care, ad esempio, 7 x 8 la tavoletta riceverà un movimento trasversale ed uno longitudinale in modo da presentare in corrispondenza delle aste dentate la coppia di asticcioline che si trova all'intersezione della settima colonna e dell'ottava linea, le quali, avendo rispettivamente le lunghezze di 5 e 6 faranno, in seguito ad uno spostamento dal basso all'alto automaticamente subito dalla tavoletta, avanzare le due aste dentate di 5 e 6 passi".

(1)

Nella figura 6 vediamo una di queste tavolette moltiplicatrici della calcolatrice di Bollée.

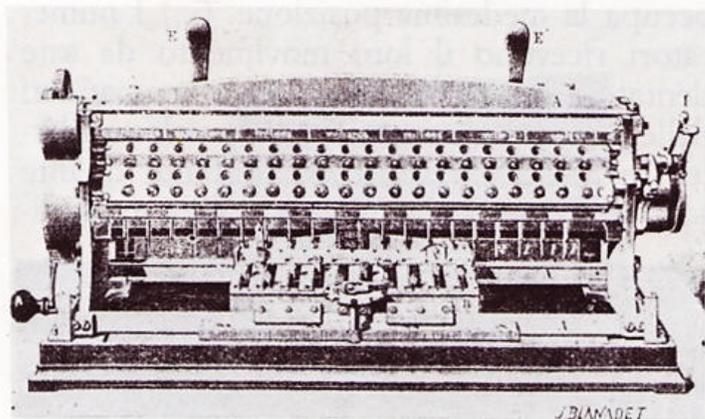


Fig. 5

Il vantaggio di disporre in un meccanismo che effettua la moltiplicazione automatica lo si riscontra nel... numero di giri di manovella che dobbiamo compiere per ottenere il risultato. Infatti con una addizionatrice è possibile moltiplicare, per esempio, 75 x 8 ma bisognerà, dopo aver impostato il numero 75, ruotare otto volte la manovella per ottenere il tanto atteso 600. Invece a partire dalla macchina di Bollée è necessaria una sola rotazione per ogni cifra che compone il numero da moltiplicare. Se cioè abbiamo il prodotto 765 x 321 bastano tre giri ed ecco il risultato.

Questa macchina, estremamente innovativa, fu presentata all'Esposizione Universale di Parigi del 1889 e si meritò una medaglia d'oro.

Di questa macchina ne furono realizzati solo tre esemplari, poiché il Bollée abbandonò la sua passione per il "calcolo" per dedicarsi ad un altro tipo di macchine: quelle da corsa. D'altronde la sua carriera di inventore era iniziata a soli tredici anni con il brevetto di

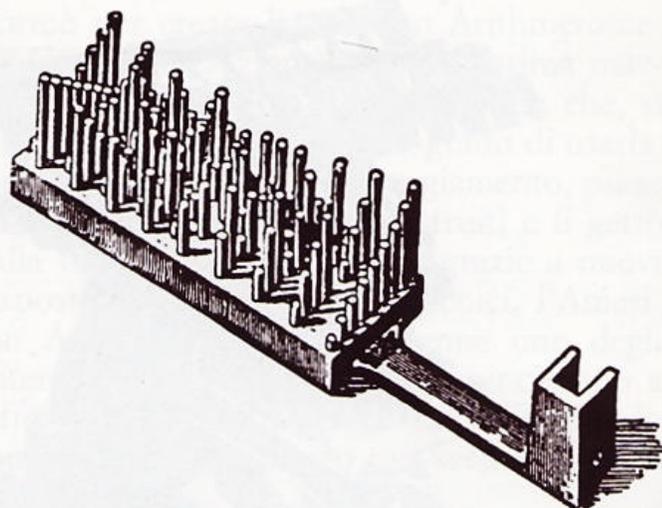
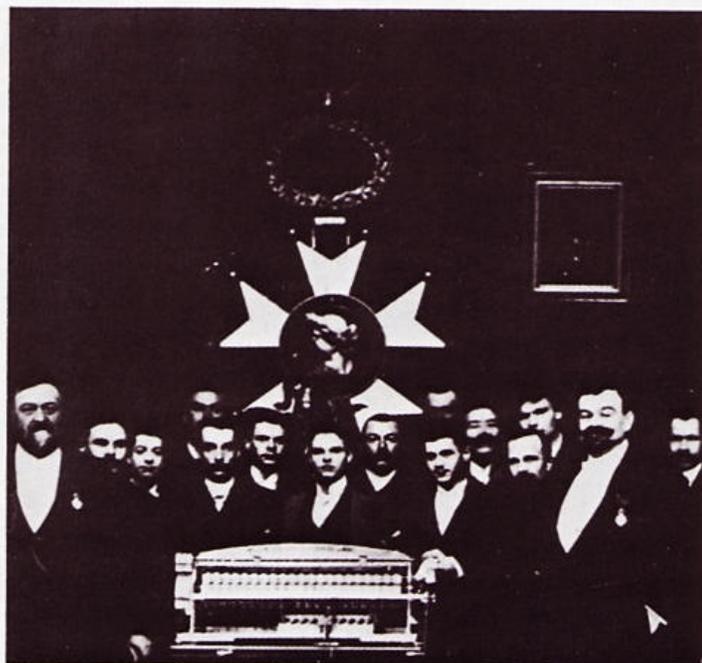


Fig. 6

una bicicletta marina inaffondabile ed era proseguita poi con la progettazione di altri diversi marchingegni come un registratore di cassa e un distributore automatico di biglietti ferroviari. (Questi due ultimi soggetti costituiscono un altro interessante capitolo dello sviluppo dell'informatica, ne parleremo in una prossima puntata).



*Nell'Esposizione Universale di Parigi del 1889 che vide il trionfo di Eiffel e della sua celebre torre un premio fu tributato a Léon Bollée che vediamo al centro della foto con la sua calcolatrice.*

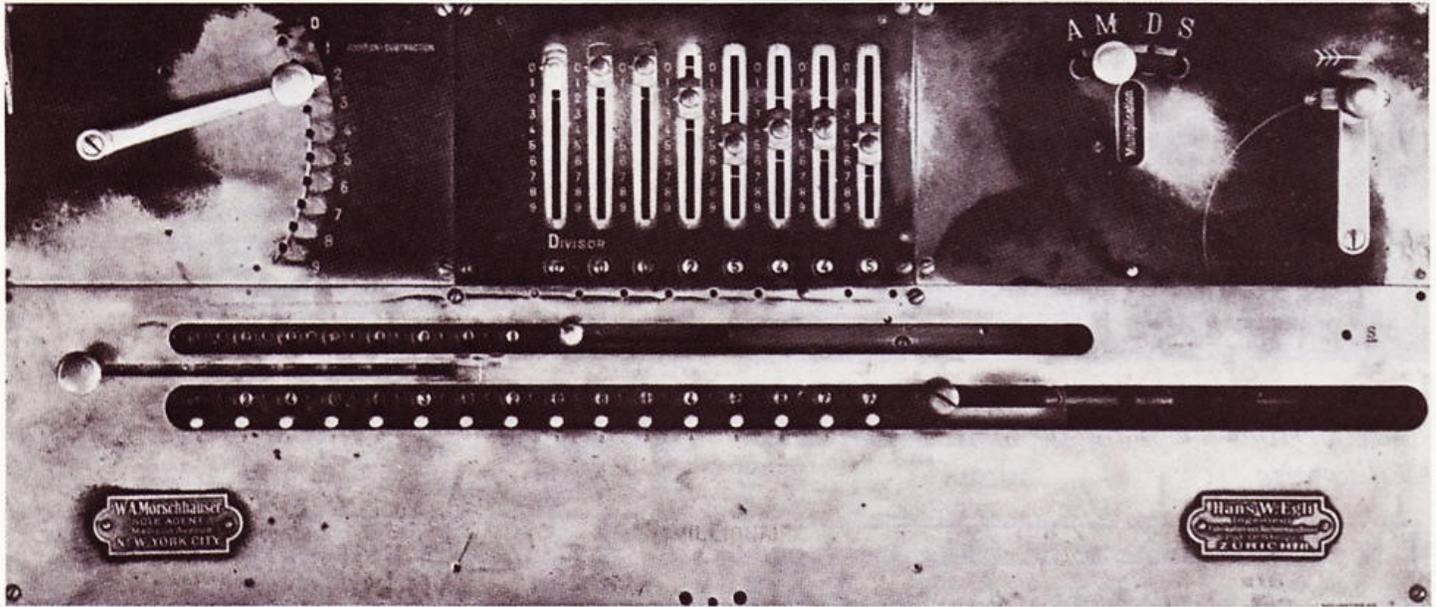


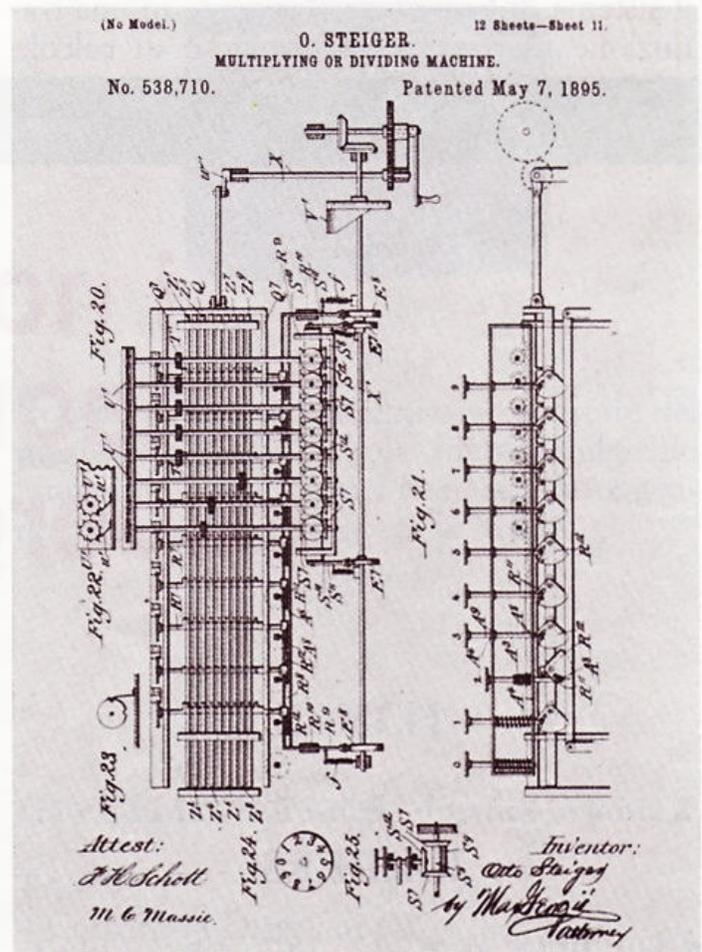
Fig. 8

Con la macchina di Bollée il calcolo automatico fa un grande passo in avanti ma, come in ogni periodo di innovazione, bisogna aspettare che altri ne intuiscano le eventuali migliorie ma soprattutto ne riescano ad individuare le possibilità commerciali. Questo avviene pochi anni dopo con la Millionaire (fig. 8) una calcolatrice progettata dallo svizzero Otto Steiger nel 1892 e prodotta su scala semindustriale dal 1895. La sua grande diffusione è testimoniata dal fatto che in quarant'anni, cioè fino al 1935, ne furono venduti 4600 esemplari (cifra che oggi appare ridicola ma che per quei tempi era da ritenersi un grosso successo).

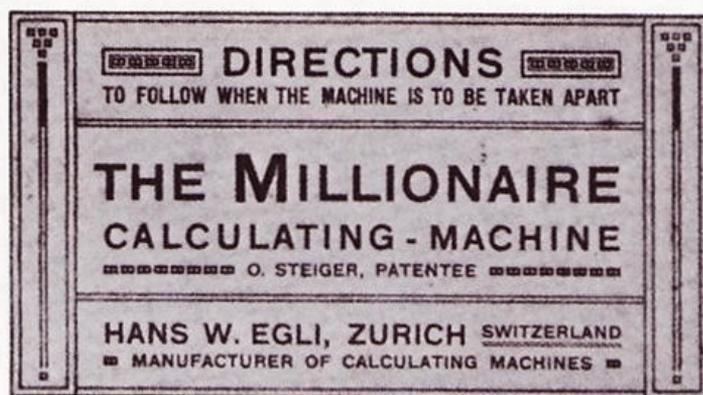
La Millionaire è basata sullo stesso principio della calcolatrice di Bollée ma sostituisce le diverse placche calcolatrici (una per ogni ordine decimale delle cifre) "con una sola placca ed un ripartitore composto di dieci lame o dieci cremagliere parallele ed equidistanti, sulle quali ciascuna colonna decimale viene a raccogliere i risultati cercati". (1)

Una semplificazione tecnica, la normale e conseguente maturazione di un'idea buona in attesa di altre migliorie, come l'introduzione, nel 1912, dell'ultima operazione aritmetica che non si poteva fare ancora direttamente: la divisione.

Questo avviene negli Stati Uniti ad opera di



Il brevetto ottenuto da Steiger per la Millionaire



*La copertina del manuale di istruzioni per la Millionaire*

J. R. Monroe anche se nel frattempo vengono scoperti ed utilizzati per le macchine da calcolo altri e completamente diversi procedimenti meccanici. Come il dispositivo di Hopkins e Moon basato su placche piane o il sistema di Seguin che "consiste in una traduzione meccanica del processo di calcolo

che va sotto il nome di "moltiplicazione ordinata". Questo procedimento segue le regole fornite dall'algebra per la moltiplicazione dei polinomi allorché si considerano le cifre di ciascuno dei fattori come i coefficienti di uno sviluppo ordinato secondo le potenze di 10". (1)

Matematica e meccanica continuano a procedere affiancate come vedremo la prossima puntata...

Byte byte a tutti

Aldo Spinelli

Note:

1. Franco Soresini, *Storia del calcolo automatico*, Roma 1977
2. René Moreau, *Ainsi naquit l'informatique*, ed. Dunod, Parigi 1981
3. *Tre secoli di elaborazione dei dati*, a cura di Roberto De Prà, IBM Italia, 1984

## i nostri magnifici supergiochi

### RAIDERS

C 16 E PLUS 4 - Tastiera - Joystick in porta 1

TASTI:

F2 : Attiva-disattiva il sonoro  
F1 : Per joystick o tastiera  
RETURN : Fuoco

Z : Sinistra  
X : Destra  
; : Alto  
/ : Basso

Il tuo compito in questa missione è di raggiungere la meta, che si trova dall'altra parte del pianeta. Dovrai volare a bassa quota con la tua



navicella, evitando le colonne e distruggendo le astronavi nemiche. Dovrai giungere a destinazione nel minor tempo possibile. Otterrai così un bonus proporzionale al tempo risparmiato.

## TIME BOMBS

**C 16 E PLUS 4 - Tastiera - Joystick in porta 1**

TASTI:

F1 : Joystick  
 F2 : Sonoro  
 BARRA : Inizio-fuoco  
 ; : Alto  
 / : Basso  
 Z : Sinistra  
 X : Destra



Sei all'interno di un labirinto ed il tuo compito è quello di riuscire a fuggire se non vuoi rimanere rinchiuso per l'eternità. Per aiutarti, appariranno sullo schermo a destra delle

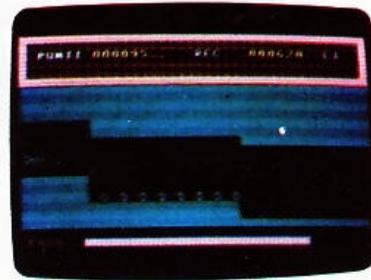
mappe di tutti i corridoi che ti illumineranno la strada per la fuga. Nelle varie stanze incontrerai degli strani esseri che cercheranno di ucciderti. Hai a tua disposizione alcune bombe: usale, se vuoi salvare la pelle.

## SKRAMBLER

**C 16 E PLUS 4 - Tastiera**

TASTI:

RETURN : Fuoco  
 £ : Bombe  
 A : Su  
 Z : Giù  
 F1 : Inizio



Vola con il tuo caccia attraverso le grotte del pianeta misterioso. Vola inarrestabilmente lungo tutti i cunicoli del pianeta, distruggendo tutto ciò che ti sbarrerà la strada.

## RANCH

**C 16 E PLUS 4 - Tastiera - Joystick in porta 2**

TASTI:

4 Cursori - 4 Direzioni

Salta con il tuo ranocchio per colorare tutta la serie di scalinate che comprendono lo schermo. Su queste scalinate dovrete fare at-

**OCCHIO RAGAZZI!**

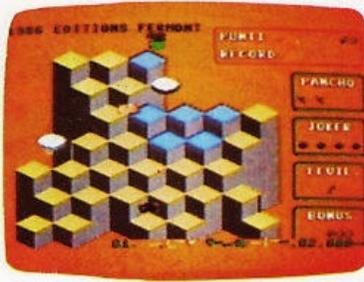
**È IN EDICOLA**

**Dzaw**  
GAMES

**A SOLE  
L.8.000**

**LA NUOVA MEGA RIVISTA  
CHE VI PROPONE ....**

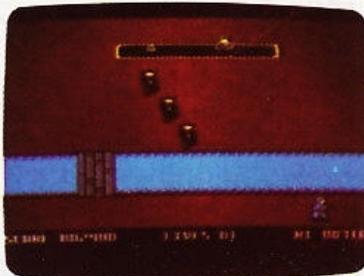
**10 Fantastici Giochi**  
*per* **Commodore 64**  
**e C 128**



tenzione agli strani esseri che vi infastidiranno cercando di colpirvi per farvi cadere dalla piramide.

## INCURSORE

C 16 E PLUS 4 - Tastiera - Joystick in porta 2



Invadi la fortezza nemica e distruggila per salvare i prigionieri che sono stati catturati. Questa è guerra! Per salvare altre vite umane, devi rischiare la tua: ti sentirai un vero eroe alla Rambo!

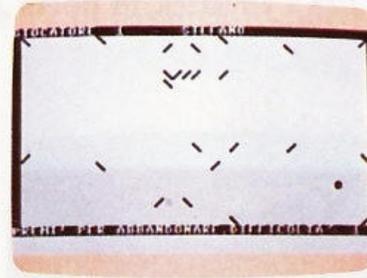
## BOUNDER

C 16 E PLUS 4 - Tastiera

### TASTI:

N : Controllo prima racchetta

M : Controllo seconda racchetta



Controlla la pallina con le due racchette a tua disposizione per colpire il bersaglio.

A questo gioco possono partecipare anche nove persone per sfide entusiasmanti in altrettanti livelli di difficoltà.

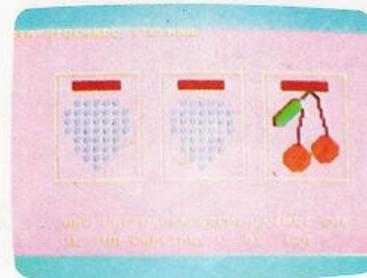
Buon divertimento!

## SLOT MACHINE

C 16 E PLUS 4 - Tastiera

### TASTI:

SPAZIO : Per tirare



Questa è il classico gioco della slot machine che ogni anno entusiasma migliaia di persone nei casinò di Las Vegas. Puoi diventare ricco da povero che sei o puoi andare in rovina pur essendo un Rock Feller.

Qui non si tratta di abilità, ma di pura fortuna: ed io ve ne auguro tanta.

## FUNGUS

CBM 64 e 128 - Joystick in porta 2



Bellissimo gioco dalla grafica multicolore e tridimensionale.

Dovete guidare l'omino nella raccolta dei funghi, saltando gli ostacoli che troverete lungo il cammino.

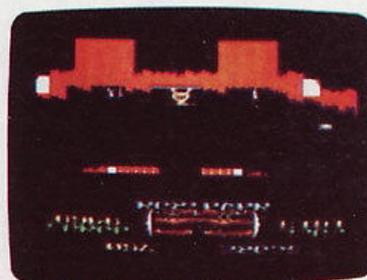
Potrete selezionare, prima di iniziare il gioco, una fra le tantissime combinazioni possibili con sette settori diversi, quattro fasi, tre livelli e sei colorazioni di scenario.

Per saltare gli ostacoli occorre spingere la leva del joystick nella direzione desiderata e premere il pulsante.

## HELIRAID

CBM 64 e 128 - Joystick in porta 2

Una missione quasi impossibile vi aspetta. Dovete salvare un gruppo di soldati che si trovano prigionieri nei sotterranei di una mega fortezza. L'unico mezzo a vostra disposizione è un potente e maneggevole elicottero. La missione sarà completata solo quando, penetrando nei vari sotterranei, avrete prelevato i soldati e, prima di uscire dalla navata centrale, distrutto il reattore nucleare per di-



sattivare tutti i raggi di protezione. Dopodiché vi dovrete posare sulla stessa piattaforma da dove eravate partiti.

Comandi: con i tasti funzione F1 ed F3 si selezionano i vari livelli di gioco e difficoltà. Premendo la barra spaziatrice, si ottiene la pausa del gioco.

## SUPER TUONO

CBM 64 e 128 - Joystick in porta 2



Guidate il vostro robot alla difesa del territorio.

I vostri nemici sono innumerevoli ed agguerritissimi, quindi non vi sarà molto facile sopraffarli. Solo con la vostra destrezza ed abilità, potrete raggiungere l'obiettivo.

Buona fortuna!!

## SCOUT

CBM 64 e 128 - Joystick in porta 2



Guidate l'omino partendo dall'accampamento base situato sulle pendici di un famoso monte. Fra il primo ed il secondo accampamento, troverete alcune tane, nelle quali vi dovrete rifugiare per difendervi dagli uccelli rapaci, naturalmente se queste tane non saranno occupate dai pericolosi orsi che frequentano il sentiero. Raggiunto il secondo accampamento, cambieranno sia lo scenario che le difficoltà.

## HELIKOPTER

CBM 64 e 128 - Joystick in porta 2

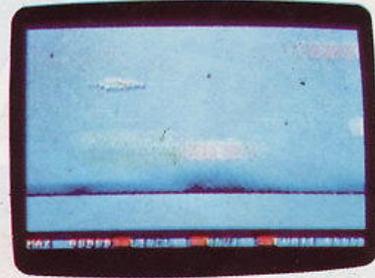


Servendovi di un elicottero, dovrete raccogliere in volo i vari paracadutisti che appariranno sullo schermo. Naturalmente la vostra missione sarà ostacolata da un nemico super armato da jet, elicotteri, missili ed altro. Dovrete quindi difendervi con molta determina-

zione, usando delle bombe e il cannoncino in dotazione.

## SPACE BLITZ

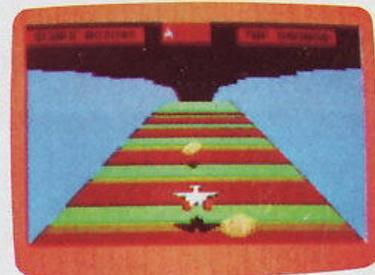
CBM 64 e 128 - Joystick in porta 1



Difendete la vostra base spaziale dall'attacco portato da numerosi nemici che riuscirete a distruggere usando il potente cannone laser montato sulla vostra astronave. La missione si riterrà conclusa quando saranno sterminati tutti gli obiettivi nemici.

## QUASAR

CBM 64 e 128 - Joystick in porta 2



Alla guida di un astro robot effettuate un viaggio interplanetario. Molte insidie vi attendono lungo il cammino, ma servendovi della vostra abilità, riuscirete a superarle, per continuare il viaggio programmato. Il gioco dispone di molti schermi, che appariranno progressivamente quando avrete superato le varie fasi.

**EHI, RAGAZZI! CHE SBALLO!!**

**È USCITA**



**A SOLE  
L.6.500**

**NOVA**  
**GAMES**

**7 GIOCHI**  
per **C 64/128**

**7 GIOCHI** per **MSX**

The logo features the word "NOVA" in large, red, 3D block letters with a white outline. Below it, "GAMES" is written in a green, italicized font on a white banner. To the left, a yellow circle contains the price "A SOLE L.6.500". To the right, there are several white stars of varying sizes, some with motion lines, suggesting excitement or a starburst effect.

**FINALMENTE IN EDICOLA**

**PER IL VOSTRO DIVERTIMENTO!**

**NON PERDETELA!!!**

# impariamo a programmare in assembler

## 11

commodore 64 - commodore 64 - commodore 64 - commodore 64 - commodore

Dopo aver realizzato ed analizzato in dettaglio il programma che esegue una moltiplicazione a 8 bit traducendo direttamente in codice l'algoritmo considerato, possiamo fare una considerazione importante: la programmazione effettiva richiede una stringente attenzione ai dettagli cosicché la lunghezza del

programma può essere ridotta con conseguente miglioramento della velocità di esecuzione.

Alla luce di questa considerazione proviamo, ora, realizzare un programma migliorato dello stesso algoritmo visto.

Una delle operazioni che consuma istruzioni

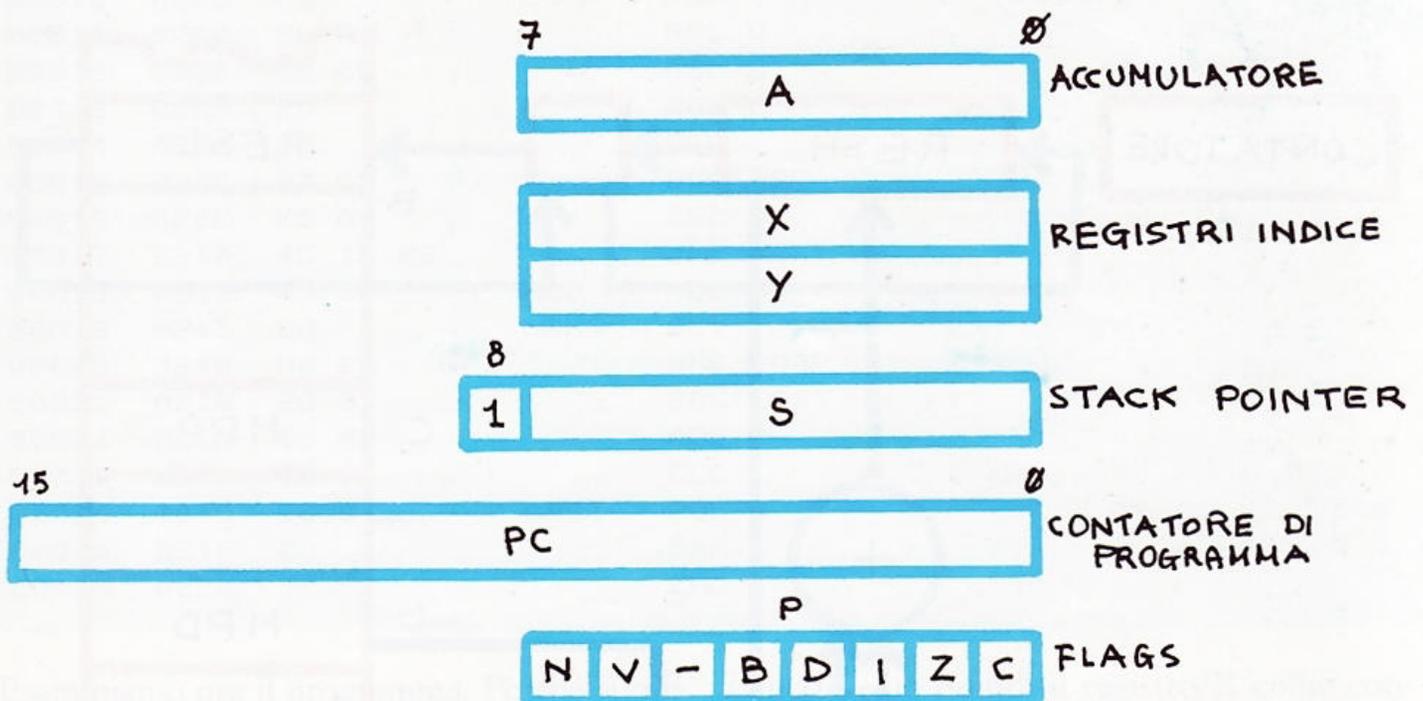


Fig. 1

e tempo è sicuramente lo scorrimento del risultato e del moltiplicatore.

Un "espediente" convenzionale utilizzato nell'algoritmo della moltiplicazione è basato sulla seguente osservazione: ogni volta che il moltiplicatore è fatto scorrere di una posizione di bit a destra, diventa disponibile sulla sinistra una posizione di bit. Contemporaneamente è possibile osservare che il primo risultato (o prodotto parziale) utilizzerà, al massimo 9 bit. Dopo il successivo scorrimento di moltiplicazione, la dimensione del prodotto parziale aumenterà ancora di un bit. In altre parole è possibile riservare, inizialmente, una locazione di memoria per il prodotto parziale e puoi utilizzare la posizione di bit che è stata liberata del moltiplicatore in virtù del suo scorrimento.

Si sta ora facendo scorrere il moltiplicatore a sinistra. Si libererà una posizione di bit sulla destra. Si fa entrare il bit più a destra del prodotto parziale in questa posizione di bit appena liberata. Consideriamo ora il programma.

Prima di listarlo consideriamo anche l'utiliz-

zazione ottima dei registri. I registri interni del 6510 appaiono in figura 1. X è meglio utilizzato come un contatore. Questo sarà utilizzato per contare il numero di bit spostati. L'accumulatore (purtroppo) è l'unico registro interno che può essere fatto scorrere. Per migliorare l'efficienza, si dovrebbe immagazzinare in esso il moltiplicatore oppure il risultato.

Quale dei due metteremo nell'accumulatore? Direi sicuramente il risultato poiché il risultato deve essere sommato al moltiplicando ogni volta che scorre fuori un 1 ed essendo che il 6510 somma sempre soltanto qualcosa all'accumulatore.

Gli altri numeri devono risiedere nella memoria (vedi figura 2). A e B conserveranno il risultato. A conserverà la parte alta del risultato e B quella bassa. A è l'accumulatore e B una locazione di memoria, preferibilmente in pagina 0. C conserverà il moltiplicatore (sarà una locazione di memoria) e D conserva il moltiplicando (anche in questo caso una locazione di memoria).

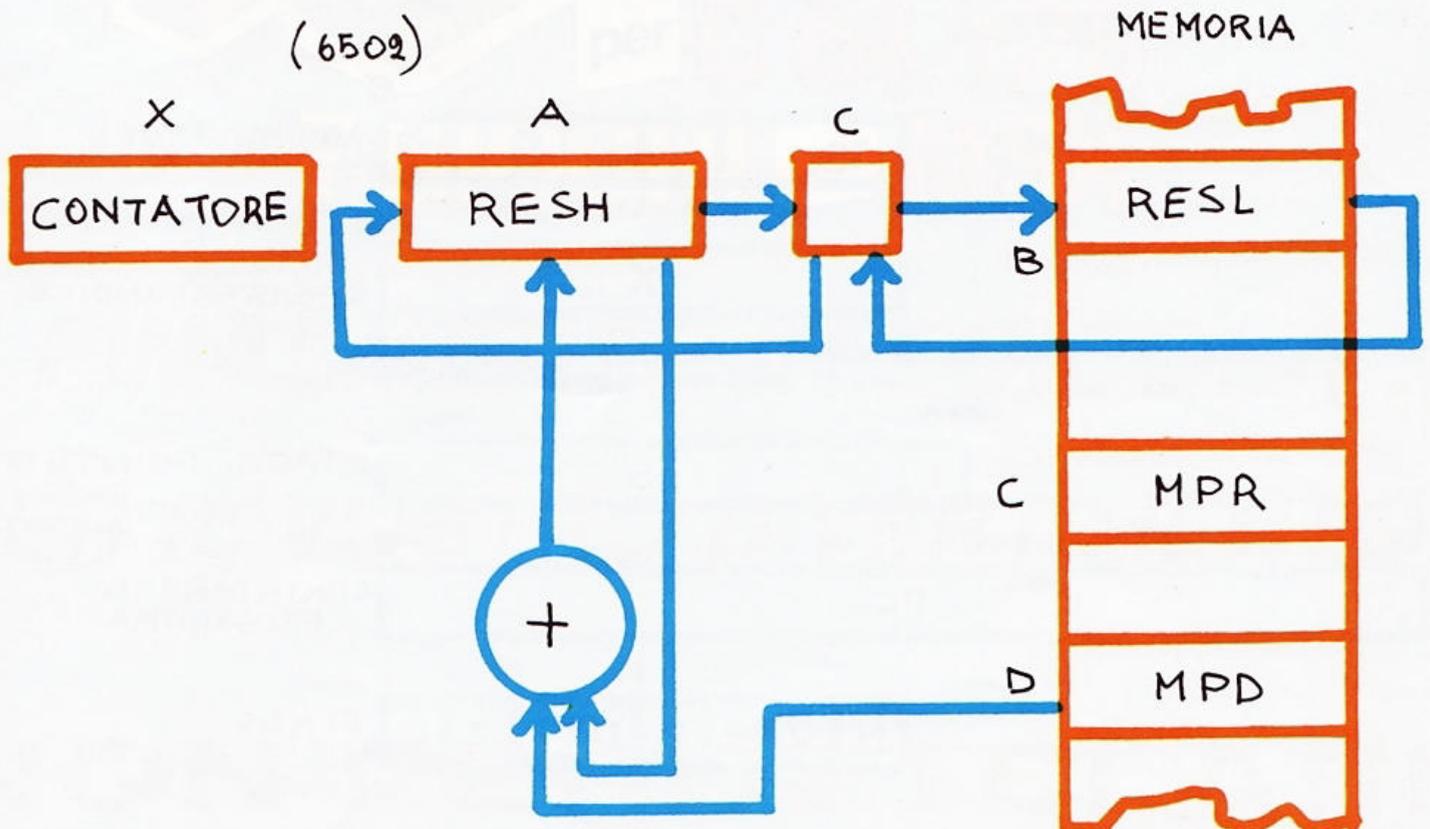


Fig. 2

## Listato 1

```

MULT      LDA  #0      INIZIALIZZA IL RISULTATO A ZERO (ALTO)
          STA  B        INIZIALIZZA IL RISULTATO (BASSO)
          LDX  #8      X E' IL CONTATORE DI SCORRIMENTI
LOOP      LSR  C        SCORRE MPR
          BCC  NOADD
          CLC          CARRY ERA UNO. VIENE AZZERATO
          ADC  D        A=A+MPD
NOADD     ROR  A        SCORRE IL RISULTATO
          ROR  B        BIT INSERITO IN B
          DEX          DECREMENTA IL CONTATORE
          BNE  LOOP    ULTIMO SCORRIMENTO?

```

## Listato 2

LINE#	LOC	CODE	LINE
00001	0000		START
00002	0000		*=\$0
00003	0000		B       *=\$+1
00004	0001		Q       *=\$+1
00005	0002		D       *=\$+1
00006	0003		*=\$200
00007	0200	A0 08	DIV     LDY #8
00008	0202	38	SEC
00009	0203	E5 02	SBC D
00010	0205	08	LOOP   PHP
00011	0206	26 01	ROL Q
00012	0208	06 00	ASL B
00013	020A	2A	ROL A
00014	020B	28	PLP
00015	020C	90 05	BCC ADD
00016	020E	E5 02	SBC D
00017	0210	4C 15 02	JMP NEXT
00018	0213	65 02	ADD     ADC D
00019	0215	88	NEXT   DEY
00020	0216	D0 ED	BNE LOOP
00021	0218	B0 03	BCS LAST
00022	021A	65 02	ADC D
00023	021C	18	CLC
00024	021D	26 01	LAST   ROL Q
00025	021F	00	BRK
00026	0220		END

Esaminiamo ora il programma. Poiché A e B conservano il risultato e devono essere inizializzati al valore 0 dobbiamo eseguire:

```

MULT LDA #0
STA B

```

Utilizzeremo quindi il registro X come contatore di scorrimento e sarà inizializzato ad 8:

```

LDX #8

```

Siamo ora pronti per iniziare il ciclo di molti-

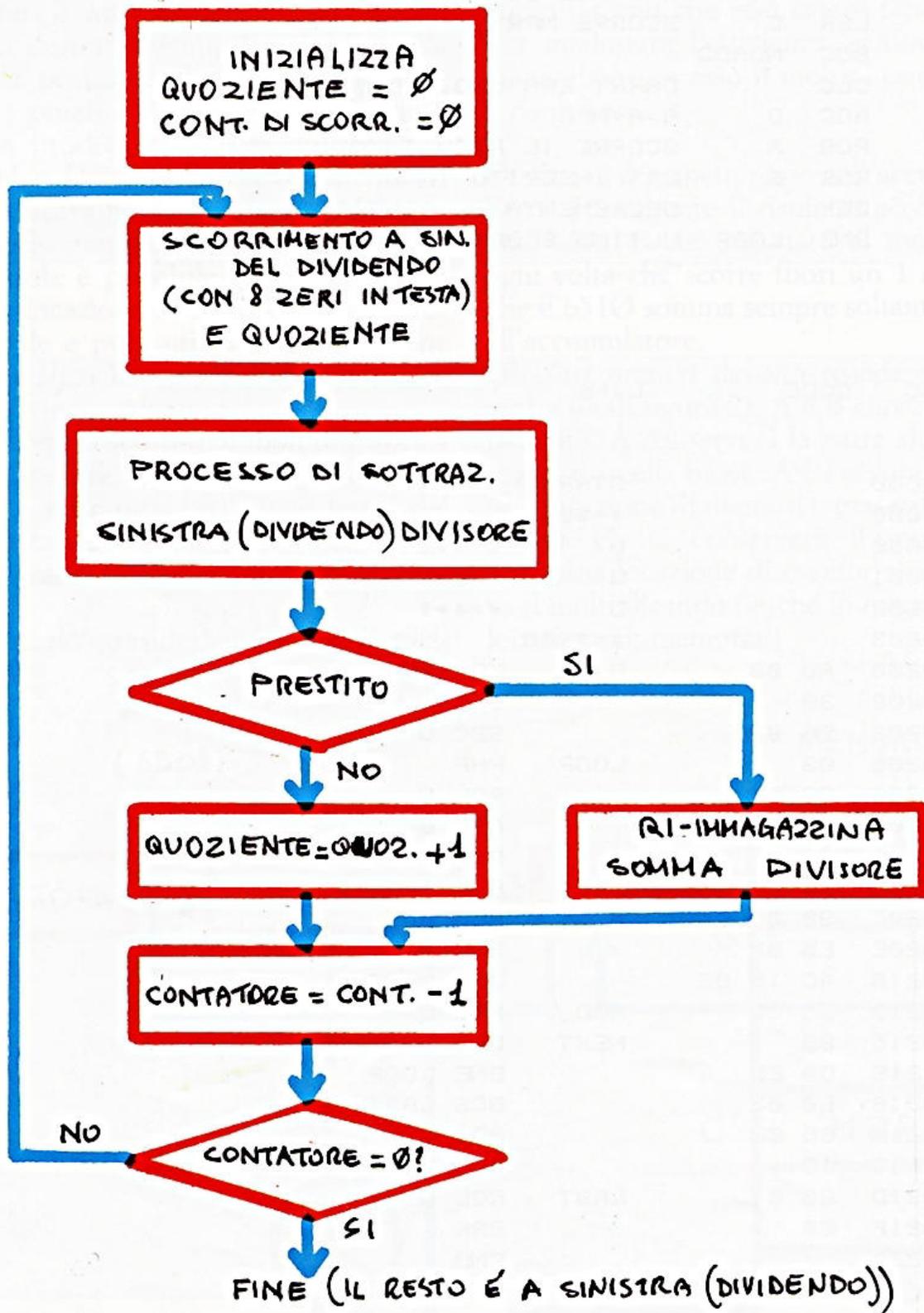


Fig. 3



plicazione principale come nel vecchio programma. Faremo scorrere prima il moltiplicatore, quindi si verificherà il bit carry che conserva il bit più a destra del moltiplicatore caduto fuori:

LOOP LSR C  
BCC NOADD

Ora si fa scorrere il moltiplicatore a sinistra (nel vecchio programma lo facevamo scorrere a destra). Questo è equivalente al precedente algoritmo in quanto l'operazione di addizione è commutativa.

Le possibilità sono due: se il carry era 0 si andrà a NOADD altrimenti si procederà così:

CLD  
ADC D

In pratica abbiamo azzerato il carry ed abbiamo sommato il moltiplicando all'accumulatore. (L'accumulatore conserva i risultati, 0 fino a questo momento).

Ora facciamo scorrere il prodotto parziale:  
NOADD ROR A

ROR B

Il prodotto parziale in A è fatto scorrere a destra di un bit. Il bit più a destra cade nel bit carry. Il bit carry è catturato e ruotato nel registro B che conserva la parte bassa del risultato.

Dobbiamo ora verificare se l'operazione è conclusa:

DEX  
BNE LOOP

Se esaminiamo il presente programma e lo confrontiamo col precedente ci accorgeremo che questo è composto dalla metà delle istruzioni impiegate precedentemente con conseguente miglioramento della velocità di esecuzione.

Questo mostra l'importanza del selezionamento corretto dei registri che contengono l'informazione.

Un progetto diretto originerà un programma

sicuramente funzionante non certamente ottimizzato. Ne deriva che è molto importante utilizzare i registri disponibili e le locazioni di memoria nel modo migliore possibile. L'esempio visto illustra un approccio razionale alla selezione dei registri per ottenere la massima efficienza.

Risolta la moltiplicazione proviamo a considerare la divisione binaria. L'algoritmo della divisione è analogo a quello utilizzato per la moltiplicazione. Il divisore è successivamente sottratto dai bit di ordine elevato del dividendo. Dopo ogni sottrazione, il risultato è utilizzato al posto del dividendo iniziale. Il valore del quoziente è contemporaneamente aumentato di 1 ogni volta. Eventualmente è negativo. Questo è definito un eccesso. Si deve quindi immagazzinare il risultato parziale riaggiungendo il divisore ad esso. Naturalmente il quoziente deve essere contemporaneamente decrementato di 1. Il quoziente e dividendo sono poi fatti scorrere di una posizione di bit e l'algoritmo è ripetuto.

Questo metodo è chiamato "metodo a ri-immagazzinamento". Una variazione di questo metodo che produce un miglioramento di velocità di esecuzione è detto metodo "senza ri-immagazzinamento".

Descriverò ora la divisione senza rimemorizzazione per un dividendo a 16 bit ed un divisore di 8 bit. La figura 4 riporta il diagramma di flusso di questo programma. Il dividendo è contenuto nell'accumulatore (parte alta) e nella locazione di memoria O, qui indicata con B. Il risultato è contenuto in Q (locazione di memoria 1). Il divisore è contenuto in D (locazione di memoria 2). Il risultato sarà contenuto in D ad A (A conterrà il resto).

Con questo abbiamo concluso anche questo mese, il prossimo articolo tratterà le operazioni logiche e le subroutine, non mancatelo; ciao a tutti.

# BUON LAVORO!!

# impariamo a usare il computer 12

Prima di riprendere a parlare di bits, bytes, linguaggio macchina, videogames, desideravo salutare tutti voi lettori augurandovi una buona lettura, naturalmente sempre in compagnia dell'amico GO GAMES.

Detto questo, rituffiamoci senza ulteriori indugi nel discorso intrapreso lo scorso mese. Riassumendo, avevamo introdotto il concetto di codice macchina partendo da un punto di vista più concettuale che applicativo. Ci eravamo quindi soffermati sui problemi pratici inerenti alla sua manipolazione, illustrando infine molto genericamente le caratteristiche salienti dell'equivalente linguaggio assembler, contraddistinto da una forma espressiva letterale molto più vicina alla mentalità umana.

D'ora in avanti tenderemo a distaccarci progressivamente da questa visione teorica per addentrarci sempre più in nozioni decisamente applicative. In poche parole, impareremo nel vero senso dell'espressione ad usare questo nostro computer. Ovviamente non tutti i calcolatori presentano le medesime peculiarità, per cui saremo costretti a considerare uno in particolare come modello di riferimento. La scelta è caduta sul COMMODORE 64 per vari motivi: innanzitutto perché G.G. è prevalentemente indirizzata ai suoi utenti, secondariamente perché esso rappresenta il più diffuso home-computer sul territorio nazionale, ed in ultima analisi in base a considerazioni tecniche che rendono questo calcolatore molto interessante sotto l'aspetto didattico.

Il microprocessore in esso contenuto, denominato 6510, altro non è che una versione

evoluto del famoso 6502, caratterizzato da un codice macchina sufficientemente versatile ma non particolarmente complesso, al contrario di altri suoi simili quali ad esempio lo Z80.

Il nostro discorso tecnico si articolerà su tre livelli di approfondimento. Tratteremo nell'ordine quelle nozioni di base comuni a tutti i calcolatori, quindi faremo riferimento più specificatamente alle istruzioni ed alle applicazioni del set di istruzioni del microprocessore 6502, per arrivare infine all'implementazione diretta sul nostro COMMODORE 64 di quanto appreso volta per volta. Naturalmente ci appoggeremo ad esso per realizzare tutti gli eventuali esempi illustrativi di programmazione.

Mi accorgo soltanto adesso di essermi probabilmente dilungato troppo in considerazioni, per cui passo immediatamente ad introdurre l'argomento di oggi. Parleremo della nozione d'indirizzamento della memoria, struttura di base indispensabile all'identificazione di ogni sua locazione.

Sarete senza dubbio a conoscenza del fatto che il COMMODORE 64 possiede 64 Kbytes di memoria, da cui il suo nome. Tale valore rappresenta il numero totale delle singole locazioni contenute nel computer. Ognuna può essere idealmente considerata come una scatoletta contenente un solo carattere, una sola informazione.

Avendo a disposizione oltre sessantaquattromila scatolette separate, il computer deve essere necessariamente in grado di strutturare un sistema di localizzazione, in modo tale da poter ritrovare agevolmente separate infor-

mazioni in qualunque momento. Questo sistema consiste nell'assegnare a ciascuna scatoletta un personale indirizzo, concettualmente simile a quello di casa vostra; tale indirizzo viene infatti utilizzato per il ritrovamento di una particolare casa in una città affollata, per l'invio di corrispondenza, oppure per il prelevamento di un pacchetto a voi destinato.

Il computer, come noi, invia informazione e si sposta da un luogo ad un altro, utilizzando il suo proprio sistema d'indirizzamento. Questo sistema è senza dubbio più semplice del nostro, in quanto consiste unicamente nel numerare progressivamente da 0 a 65535 ogni singola locazione di memoria; per noi tutto ciò non risulterebbe molto semplice da ricordare, ma per il computer rappresenta esattamente il suo logico modo di agire. Queste scatolette numerate possono essere assimilate a delle caselle postali; nel caso abbiate depositato qualcosa nella casella numero uno, questo vi resterà fino ad un vostro successivo intervento. In ogni scatoletta potrà essere posto un unico oggetto alla volta: iserendone uno nuovo, si perderà automaticamente e per sempre il precedente contenuto.

Tali scatolette, che d'ora in avanti chiameremo bytes o locazioni di memoria, sono a loro volta suddivise in otto cellette assimilabili ad un interruttore a due posizioni. Esse potranno pertanto risultare accese o spente, e niente di più. Lo stato di accensione viene convenzionalmente rappresentato per mezzo della cifra "1", mentre quello di spegnimento sarà identificato attraverso uno "0". Ognuna di queste otto cellette, la cui denominazione tecnica è BIT (forma contratta dell'espressione inglese BInary digiT, in italiano cifra binaria), occupa un posto ben preciso nella locazione di memoria alla quale appartengono.

Questo posto costituisce un vero e proprio ordine di grandezza, assolutamente equivalente alle unità, decine, centinaia, migliaia del sistema di numerazione decimale adottato dalla maggior parte degli esseri umani nella vita di tutti i giorni. Una locazione di memoria è composta pertanto da una sequenza di otto bits contigui, ciascuno dei quali può contenere unicamente la cifra "0" o la cifra "1".

Calcoliamo adesso il numero di tutte le configurazioni che può assumere una locazione di memoria, in altre parole tutte le combinazioni numeriche ottenibili attraverso otto cifre binarie. Il calcolo è molto semplice. Poiché ciascuna cifra assumere due stati, il numero che noi cerchiamo si raddoppia per ogni cifra aggiunta in più. Considerandone una sola, il risultato è ovviamente "2". Con due cifre, avremo quattro combinazioni, con tre cifre queste diventeranno otto, e così di seguito. La formula generale che adotteremo per  $n$  cifre si deduce molto semplicemente:

$$N = 2 \uparrow n$$

dove  $N$  rappresenta il numero di combinazioni possibili e  $n$  il numero di cifre considerate. Nel nostro caso particolare riferito ad una locazione di memoria abbiamo  $n = 8$ , per cui  $N = 2 \uparrow 8 = 256$ . Ogni byte può dunque immagazzinare 256 diversi valori, precisamente da 0 (tutti ed otto i bits azzerati) a 255 (tutti ed otto i bits posti ad uno).

Osservando attentamente la formula precedentemente applicata, notiamo che  $N$  si esprime in funzione di un parametro variabile ( $n$ ) e di una costante (2). Sarebbe interessante generalizzare ulteriormente tale formula esprimendo  $N$  in funzione di due parametri interi positivi. In pratica, invece di avvalerci unicamente d'interruttori a due posizioni, strutturiamo il nostro ragionamento considerando globalmente interruttori con numero di posizioni da due all'infinito. Quelli ad una sola posizione o addirittura quelli senza posizioni (assurdi...) non li consideriamo poiché inutili sotto ogni punto di vista.

La nostra formula si esprimerebbe quindi nel modo seguente:

$$N = k \uparrow n$$

dove  $N$  rappresenta il numero di combinazioni possibili,  $k$  il numero di posizioni assumibili dai nostri interruttori ed  $n$  la quantità di questi ultimi. Non notate niente di familiare assegnando a  $k$  il valore 10? Certo che sì! Non abbiamo fatto altro che definire il classico sistema di numerazione decimale.

Ogni cifra può infatti assumere dieci valori differenti, quelli da 0 a 9, che altri non sono che le posizioni dei nostri interruttori.

Domanda: quanti numeri si possono esprimere con tre cifre decimali?

Risposta:  $N = 10^3 = 1000$

In effetti, con tre cifre decimali possiamo rappresentare tutti i numeri da 0 a 999, ovvero mille valori differenti.

Ritornando al nostro calcolatore, prenderemo in considerazione soltanto particolari tipi d'interruttore. Per la precisione, ci occuperemo di quelli classici a due posizioni (bits), nonché di un nuovo tipo a sedici posizioni che si rivelerà particolarmente utile in tutte le procedure di calcolo relative all'indirizzamento della memoria (visto che alla fine siamo ritornati al punto di partenza?).

Arrivederci al mese prossimo.



# DIVERTITEVI CON I BELLISSIMI GIOCHI DI GO GAMES!!



IL NUMERO 5

**È IN EDICOLA**

LA NUOVA RIVISTA  
CON CASSETTA DI  
**ADVENTURES**

TUTTE IN ITALIANO CHE GIRANO SU  
CBM 64/128 E SPECTRUM 48 K

ADVENTURE  
**EPIC**  
3000  
GAMES

IN ITALIANO

N° 5

OTTOBRE 1986

L. 8.000

ADVENTURE GAMES PER C. 64/128 E SPECTRUM 48K

Commodore 64 e 128

**JACK BYTESON**  
Vendetta di Abel Kuyr  
(AVVENTUROSO)

**FUGA DAL LAOS**  
(GUERRA)

**HORROR**  
Posto poco tranquillo  
(HORROR)

Spectrum 48K

**MYTHOS**  
(MITOLOGICO)

**COMPUTER KILLER**  
(FANTASCIENZA)

**VIRUS DELTA**  
Ultimo atto  
(POLIZIESCO)

