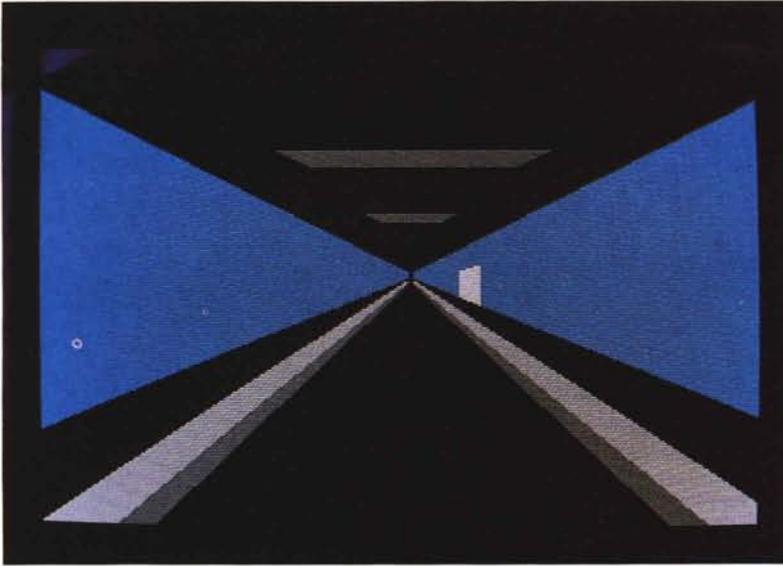


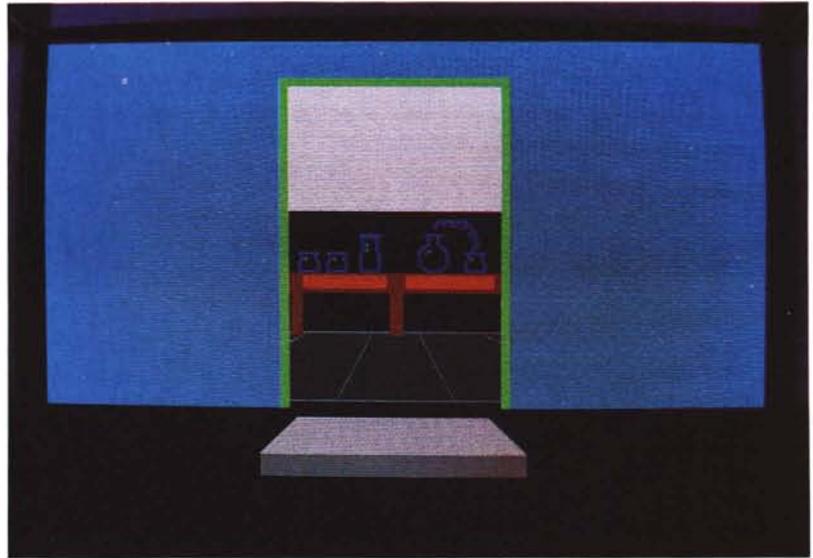
Foto 3



e quindi impostare i giusti comandi senza dover per forza scandagliare tutta la mappa.

Osserviamo le altre tre foto. Cosa rappresentano? Ho effettuato una piccola modifica alla scena nel tunnel. La visuale è inizialmente quella della foto 2, quindi una prospettiva isometrica. Paolo entra nel mezzo di trasporto (si è proprio quello) e decide quale direzione prendere. Il menzionato monoposto comincia la sua corsa. Il fondale è fisso mentre accelera. Raggiunta una certa velocità la vista cambia e si sfreccia nel tunnel con vista in prima persona (foto 3). Giunti nei pressi della destinazione si ritorna alla scena della foto 2 e il monoposto rallenta fino alla porta. Una volta fermatosi la scena cambia con quella di foto 4. La porta comincia a sollevarsi e osserviamo l'interno del compartimento. Entriamo? e la vista cambia ancora!

Foto 4



## Scuola di videogame

### Moltiplicazione degli sprite (3)

Finalmente vedo il listato sulle pagine di MC. E già perché con quel fogliaccio lungo che mi ritrovavo era facile sbagliarsi e saltare qualche istruzione... è successo!!! Sfruttando la confusione generale che si era generata sul mio tavolo il diavoletto è riuscito a metterci lo zampino e mi ha fatto confondere. Dov'è l'errore direte voi (... se non siete dei lettori attenti); le ultime parole famose: «... Locazione \$7081, nuova istruzione: JSR...». La locazione \$7081 non c'entra proprio niente; la nuova

istruzione infatti si ritrova già a partire dalla locazione \$7051 ed è proprio a questa locazione che dovevano andare i miei occhi un mese fa. Chiarito l'equivoco (almeno spero) possiamo riprendere il nostro discorso. Dunque, alla locazione \$7051 troviamo quindi la prima JSR che in questo caso salta alla locazione da \$70C0. La routine memorizzata nelle locazioni da \$70C0 a \$70CB è molto simile alle altre due routine che seguono (in ordine di locazioni), memorizzate da \$70D0 a \$70DB e da \$70E0 a \$70EB

Ce ne andiamo a spasso per il settore (niente foto...) e poi usciamo. Paolo rientra nel suo «macinino» e via... Siamo giunti così anche questo mese alle conclusioni anche se la mia promessa non è stata mantenuta per intero. Di tempo ne abbiamo ma non sprechiamolo troppo e mi rivolgo soprattutto a voi. Se non siete rapidi le vostre lettere rischiano di arrivare in ritardo e così va veramente a finire che il gioco lo faccio solo io. Per quello che riguarda la grafica vi concedo ancora un po' di tempo. Per quello che riguarda la struttura di gioco cercherò di completarla entro i prossimi due mesi. A buon intenditor poche parole.

e infatti servono per eseguire su per giù lo stesso compito (... su per giù). Esaminiamo la prima. Locazione \$70C0: LDX # \$10; nessun problema ma ricordiamoci di quanto abbiamo fatto (mettiamo # \$10 in X). L'istruzione successiva è già più impegnativa; cosa vuol dire quel "X" dopo il ben noto LDA \$7100. Cosa abbiamo messo in X? (l'ho detto 2 secondi fa). Se vi dicessi che nell'accumulatore, invece della locazione \$7100, viene caricata la locazione \$7110 mi credereste? ... cretedemi. Questa «nuova» istruzione quindi somma il valore del registro X al valore fornito dall'istruzione. Volendo fare la stessa cosa in Basic bisognerebbe scrivere:

```
A= PEEK ([locazione]+X)
```

dove [locazione] è un valore qualsiasi mentre X è una semplice variabile. Ricapitolando il registro X viene usato come «puntatore». Continuiamo la lettura del

programma. La prossima istruzione è simile a quella appena vista ma è applicata alla funzione STA, quindi in Basic sarebbe:

POKE [locazione] + X, A

Andiamo avanti anche se non abbiamo ancora afferrato a cosa serve questa benedetta routine. Nuova istruzione fresca fresca: DEX. Guarda caso questa istruzione decrementa il contenuto del registro X. La solita analogia con il Basic ci fa scrivere:

X=X-1

A questo punto troviamo un'istruzione veramente strana: BPL \$70C2. Proprio qui sta il succo di tutta la routine; si tratta di un salto condizionato (!?). Niente paura, mi appresto a spiegarvi cosa significa. Come dice la parola stessa e quindi come è intuibile, un salto condizionato è un salto che per essere effettuato ha bisogno che una determinata condizione sia verificata. In questo caso il salto, al verificarsi della condizione, avverrebbe alla locazione \$70C2 quindi il computer ricomincerebbe ad eseguire le istruzioni già viste e se non fosse del tipo «condizionato» non «uscirebbe» mai dalla routine. La condizione in questo caso è: «salta se il risultato della precedente istruzione ha prodotto un numero maggiore di zero». Quindi osserviamo il risultato prodotto dall'istruzione precedente: è sicuramente maggiore di zero, visto che è stato decrementato il registro X che prima era a \$10 (ora sarà a \$0F quindi è ben lontano dall'essere «minore di zero»). Il salto quindi in questo caso viene effettuato e le istruzioni viste prima vengono rieseguite fin quando X viene decrementato dal valore zero al valore inferiore (che per ora non diciamo qual è). In tal caso l'istruzione BPL non si farebbe scrupoli e non darebbe luogo al salto. Ci sarebbe quindi il ritorno al programma «principale» grazie all'RTS. Il gruppetto di istruzioni quindi viene rieseguito per ben 16 volte prima di «cambiare aria». A questo punto fate intervenire la logica.

Il programmino viene eseguito ogni volta con un valore di X diverso, quindi il valore che si addiziona alle istruzioni LDA \$7100, X e STA \$D000, X cambia di volta in volta. Questo vuol dire che i valori contenuti dalla locazione \$7100 alla \$7110 vengono trasferiti alle locazioni dalla \$D000 alla \$D010, ma forse ci eravate arrivati già da soli. Vogliamo capire a cosa serve tutto questo? Dalla locazione \$D000 alla \$D010 troviamo i registri delle coordinate degli otto sprite che in questo caso verrebbero alterate. Ricordate quando dicevo che questa routine è ripetuta in memoria tre volte ma con una piccola modifica? Se andia-

mo ad osservare le due routine seguenti ci accorgiamo che l'unica cosa che cambia tra loro, oltre la posizione in memoria e quindi i salti delle BPL, è il valore che segue la LDA, che nei tre casi è, rispettivamente, \$7100, \$7120 e \$7140. Questi tre valori delle routine corrispondono quindi a tre cambiamenti di coordinate, che se effettuati nella stessa pennellata video danno origine alla moltiplicazione degli sprite! ( $8 \times 3 = 24$ ). Ovviamente questi tre cambiamenti devono essere effettuati al momento opportuno e quindi le tre routine sono gestite dalle tre rispettive routine IRQ che fanno in modo che il cambiamento si verifichi ad una ben determinata posizione del pennello elettronico. Se infatti da bravi bambini vi andate a guardare il resto del programma che non abbiamo ancora esaminato capirete come si svolge il truccetto. Molti di voi avranno finalmente afferrato come funziona questo benedetto moltiplicatore di sprite e si meritano un «bravo». Agli altri devo dare appuntamento alla prossima puntata perché per questo mese non c'è proprio più tempo. Vi ricordo comunque che la prossima volta cominceremo anche un nuovo argomento. Saluti a tutti.

## Megaposta

*Gentilissimo Marco, dopo aver letto la Megaposta sul numero 73/aprile, ho deciso di inviare due righe per rispondere in maniera adeguata ad Alessandro Merolli di Roma.*

*Carissimo Alessandro, qui nessuno si crede di essere un EINSTEIN!!! Il fatto che tu non riesca a capire e/o risolvere un gioco matematico non implica che tutti siano degli incapaci, ma è soltanto indice della tua insufficiente capacità d'analisi dei problemi.*

*Come hai potuto constatare la soluzione era alquanto semplice! Mi scuso per aver abusato di questa rubrica per una comunicazione personale, ma era l'unico modo possibile per contattare Alessandro.*

*Gianni Zambon, Treviso*

Aspettavo la tua replica Gianni... ma non sognatevi di continuare la discussione su queste pagine...

Per questo mese la Megaposta è stata tutt'altro che mega, ma purtroppo non ho il tempo per rispondere alle altre lettere e l'unica che mi sembrava «degnata di urgenza» l'ho pubblicata. Scusate.

Il più grande e potente foglio elettronico per personal computer della nuova generazione, è anche il più veloce, il più facile, il più bello.



Quando vedrai di cosa è capace Microsoft Excel ti chiederai come hai potuto farne a meno finora. Microsoft Excel non è solo il miglior foglio elettronico esistente: è meraviglioso. Adesso, parla italiano in due versioni; per personal computer MS-DOS e per Apple Macintosh. Sempre più meraviglioso.

### Il miglior strumento analitico

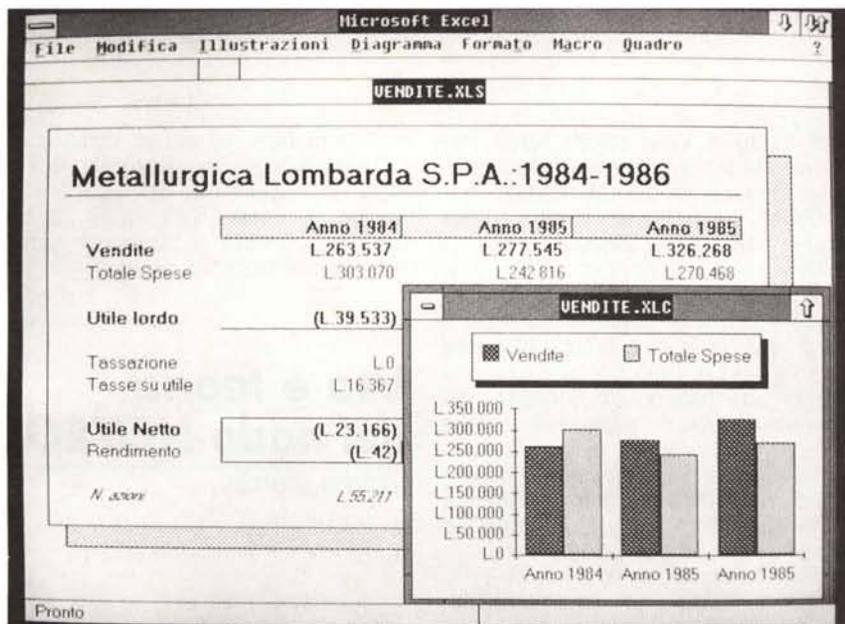
Con Microsoft Excel le dimensioni del foglio non hanno più praticamente confini: ben 256 colonne per 16384 righe per sfruttare tutta la potenza dei personal computer della nuova generazione, creare modelli molto grandi e offrire pieno supporto alle espansioni di memoria: poi Microsoft Excel ti impressionerà per la sua velocità e per la sua intelligenza nel ricalcolare il foglio solo dove e quando serve e per la possibilità di richiamare sullo schermo più fogli contemporaneamente e stabilire collegamenti tra loro. E se credi che sia solo un foglio elettronico ti sbagli: sono state integrate anche potenti funzioni di database.

### Il miglior strumento per la presentazione

Scegli adesso uno dei tanti



# Excel delle meraviglie parla italiano



caratteri disponibili, mescola sul foglio i bordi, le ombreggiature o le sottolineature.

E soprattutto usa a tuo piacere il colore e dai spazio alla tua fantasia creativa.

Per darti un'idea delle illimitate capacità grafiche di Microsoft Excel, ti segnaliamo solo che ha ben 44 tipi di grafici predefiniti con infinite possibilità di personalizzazione.

Inoltre Microsoft Excel ti consente di visualizzare sullo schermo il tuo grafico insieme al foglio sul quale stai lavorando. Vuoi vedere come sarà il risultato della pagina stampata?

Semplice: richiamala con la funzione "anteprima", controlla sul video e dai "l'ok si stampi".

## Il miglior strumento per la personalizzazione

Hai una particolare antipatia per le macro? Puoi tirare un sospiro di sollievo. Il registratore di macro Excel registra le operazioni man mano che le esegui: e la macro è subito fatta. Non solo, con Microsoft Excel puoi personalizzare completamente l'interfaccia utente o creare un ambiente di lavoro veramente tuo. E se hai delle ambizioni scientifiche puoi contare su una gestione di matrici davvero unica.

## In diretta dal passato al futuro

Con Microsoft Excel ogni altro foglio

elettronico ti sembrerà subito vecchio e potrai finalmente sfruttare in pieno tutta la potenza dei computer della nuova generazione. Microsoft Excel, infatti, ti porta verso il terzo millennio. L'interfaccia grafica Windows di Microsoft Excel è identica a quella che adotterà il sistema operativo del futuro, l'OS/2. Così, quando avrai imparato ad utilizzare le finestre, i menu a tendina e tutte le caratteristiche grafiche di un'interfaccia sofisticata come Windows sarai in grado di utilizzare tutti i programmi che gireranno sotto OS/2 e che sfrutteranno la sua interfaccia Presentation Manager. Ma non preoccuparti per ciò che hai fatto fino ad oggi. Microsoft Excel ha una completa compatibilità con i fogli elettronici della passata generazione e ritraduce le loro macro. Inoltre consente anche un collegamento dinamico tra i diversi programmi. Ma una cosa soprattutto ti stupirà di Microsoft Excel, la facilità con la quale imparerai ad usarlo. Ci sarebbero mille altre cose da dire su Microsoft Excel, ma a questo punto perchè non ci scrivi o non ci telefoni? Riceverai tutte le informazioni e ti invieremo subito una dettagliata documentazione.

*Per informazioni:*

*Microsoft S.p.A.*

*Milano Oltre - Palazzo Tiepolo*

*20090 Segrate (MI)*

*Via Cassanese, 224*

*Tel. 02/2107.201*

**Microsoft**

*Il software del tuo successo*