

Corso alternativo di programmazione

Dodici modi per costruire una Tabella Pitagorica

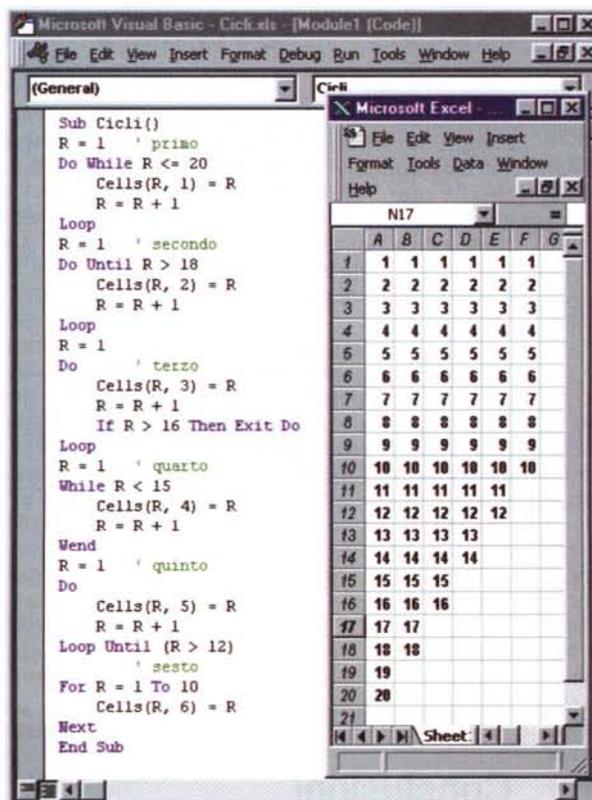
di Francesco Petroni

Se avete letto il riquadro introduttivo avrete compreso le finalità di questi articoli, che riguardano lo studio e l'approfondimento delle tecniche della programmazione e delle varie tecnologie che permettono lo sfruttamento di tali tecniche, e che non riguardano assolutamente la soluzione di problematiche applicative reali, "rivendibili" in applicazioni commerciali.

La prima "problematica applicativa" è addirittura la costruzione di una Tabella Pitagorica, argomento astrattissimo che però può essere affrontato in maniera elementare dal punto di vista del programmatore.

Realizzeremo la nostra Tabella Pitagorica in una dozzina di maniere differenti, usando diversi linguaggi e diverse situazioni ambientali: ambiente Windows, ambiente Internet, tecnologie Client/Server, tecnologia COM, ecc.

Per quanto riguarda i linguaggi di programmazione, useremo prevalentemente il Visual Basic ed i suoi vari dialetti, con qualche "puntatina" in altri linguaggi, nei casi in cui la tec-



dell'autore, anche se in realtà ciascuna variante ha propri ambiti applicativi ottimali. Qui vediamo 6 modi per contare da 1 a 20 usando il Visual Basic for Application di Excel, il cui foglio viene usato per la visualizzazione dei numeri.

nologia che si sta usando lo richiedesse, ad esempio, già in questo primo articolo, il JavaScript.

Enunciazione del problema

Dunque dobbiamo realizzare una Tabella Pitagorica.

Dal punto di vista "matematico" una Tabella Pitagorica corrisponde ad una matrice bidimensionale contenente un certo numero di righe e colonne (normalmente è una matrice 10 per 10 oppure 12 per 12), che, se calcolata con il computer, può avere qualsiasi dimensione e viene prodotta in milionesimi di secondo.

In tutti gli ambienti operativi è possibile inserire tabelle.

Sei modi per contare da 1 a 20. I moderni linguaggi di programmazione sono molto evoluti al punto che per raggiungere uno stesso obiettivo si possono praticare numerose strade alternative. Spesso la scelta di una strada rispetto a quella alternativa dipende dall'abitudine

MS Visual Basic dispone del componente griglia, anzi di griglie ce ne ha ben quattro. Noi useremo la FlexGrid perché è la più facile da caricare via codice.

Un foglio di MS Excel è esso stesso una grande tabella con milioni di celle, che potremo utilizzare facilmente per visualizzare i nostri numeri.

Word ha la tabella che serve per impaginare in modo ordinato dei dati, e che può essere usata per vedere i

numeri pitagorici.

Anche il linguaggio che si usa in Internet, l'HTML, dispone di un tag "<Table>" che serve per creare una tabella sulla pagina. Anzi, fino all'avvento del DHTML, la Table, con tutti i suoi sottotag, costituiva l'unico modo per organizzare in zone una pagina.

Quindi nessun problema per quanto riguarda l'impaginazione della nostra Tabella Pitagorica.

Per quanto riguarda invece il suo

"calcolo" occorre eseguire due loop, due cicli di calcolo, il primo che scorre tutte le righe e il secondo che, a parità di riga, scorre tutte le colonne. Il calcolo vero e proprio è la semplice moltiplicazione tra l'indice che viene usato per scorrere le righe e quello che viene usato per scorrere le colonne.

Nella figura 0 vediamo un piccolo campionario di cicli, in pratica sei modi alternativi per contare da 1 a 20 (poi a 18, a 16, ecc.), realizzato con VBA

Riprendo, dopo un anno di meritato riposo, che segue quindici anni di collaborazione ininterrotta, a scrivere articoli per MC-microcomputer. Con l'occasione saluto e ringrazio tutti i vecchi lettori, alcuni dei quali hanno continuato ad inviarmi posta, e porgo il più sincero benvenuto ai nuovi lettori che spero siano sempre più numerosi. Con gli amici della redazione abbiamo deciso che il mio compito sarà quello di parlare di programmazione, cercando però di parlare in modo alternativo rispetto al modo in cui ne parlano le decine di riviste specializzate, le centinaia di libri e di manuali e le migliaia di siti Internet destinati ai professionisti della programmazione.

In questo riquadro cercheremo di spiegare il significato che diamo al termine "alternativo", che abbiamo associato alla parola corso, e daremo le necessarie indicazioni su come si svilupperà questa prima serie di articoli.

Due premesse, la prima corta e la seconda lunga.

La prima: sabato 8 gennaio 2000 sulla prima pagina di Repubblica è apparso un articolo di Umberto Eco dedicato alle sfide del 2000 e che sostanzialmente parla di Internet ("chi non naviga è perduto", "il mondo è diviso in classi, chi naviga e chi no") e di Programmazione ("se si insegna ad un bambino un linguaggio informatico, questo esercizio logico lo renderà padrone e non schiavo del computer").

Quindi la programmazione, o meglio i principi logici e formali alla base della programmazione, deve essere intesa, secondo Eco, come bagaglio culturale indispensabile di ognuno di noi al punto che sarebbe bene che vi provvedessero le scuole tradizionali.

La seconda premessa: nella mia vita professionale ho conseguito, sostenendo circa una decina di esami (in realtà ne sono sufficienti quattro), il titolo di MCSDB, Microsoft Certified Solution Developer, il titolo di MCSB, Microsoft Certified Site Builder, ed infine quello di Microsoft Certified Trainer. Questo significa che sono riconosciuto ufficialmente dalla Microsoft, ed il titolo vale a livello mondiale, come "sviluppatore di soluzioni informatiche", tradizionali e per il Web, e come "docente" delle tecnologie sottostanti le problematiche di sviluppo. In particolare sono docente di problematiche relative all'architettura delle soluzioni informatiche, alla programmazione con MS Visual Basic, tutti i tipi di Visual Basic, ed alla programmazione per Internet, usando gli strumenti targati Microsoft, FrontPage, Visual InterDev, ecc.

Come Trainer ho tenuto centinaia di corsi e quindi ho conseguito un'enorme esperienza su come i programmatori, e gli aspiranti programmatori, affrontano lo studio dei linguaggi e più in generale lo studio delle tecnologie.

Ad esempio nei Corsi di Visual Basic incontro sia allievi che non hanno mai programmato, per i quali per intenderci il Visual Basic è il primo linguaggio di programmazione, sia allievi già esperti di altri

Corso alternativo di programmazione

linguaggi (ad esempio il Cobol se provengono dal mondo Mainframe o il Clipper se provengono dal mondo PC) che cercano di imparare facendo dei parallelismi (quasi sempre impossibili) con le conoscenze che già hanno, sia allievi già pratici del prodotto, in quanto lo hanno studiato in maniera autodidattica, ma che debbono completare le loro conoscenze.

È evidente che il mio comportamento varia a seconda del tipo di allievo che mi trovo di fronte. Ai primi, che tendono a confondere i concetti generali con gli elementi del VB, propongo inizialmente esercizi di programmazione "astratti" che prescindono del tutto dall'ambiente VB allo scopo di insegnare i principi fondamentali della programmazione. Con i secondi, gli allievi che si stanno riciclando, inizio un vero e proprio "braccio di ferro" per modificare il loro precedente modo di pensare e di lavorare per convertirli alla filosofia "object". Con i terzi il rapporto è ancora più stimolante, in quanto spesso mi capita di dover confrontare il loro modo di vedere le cose, stiamo parlando di allievi già esperti, con il mio, ed in alcuni casi sono io che "imparo" qualche cosa.

Il più delle volte il mio compito è comunque quello di riempire loro vuoti di conoscenza.

Il discorso diventa ancora più complesso quando si parla dei corsi di Programmazione per Internet, in cui anche la più semplice delle applicazioni è comunque un'applicazione client/server, in cui intervengono e collaborano tra di loro due pezzi di programma, ognuno dei quali svolge un compito, ed in cui bisogna prevedere e valutare condizioni "al contorno", la prima delle quali è la compatibilità del browser.

In questi casi è necessaria, oltre che la conoscenza degli strumenti di sviluppo e la conoscenza delle tecnologie coinvolte, anche la padronanza della filosofia client/server.

Tornando al nostro corso di programmazione, lo abbiamo definito alternativo proprio perché ha l'ambizione di essere destinato contemporaneamente sia ai principianti della programmazione, che trovano esercizi facilmente realizzabili (tutti i listati che presentiamo "entrano" in una videata), che li introducono in maniera diretta nel mondo della programmazione e permettono loro di sperimentare immediatamente concetti anche non del tutto banali, sia ai più esperti, che possono sia "assestare" le loro conoscenze, anche perché trovano esemplificate alcune delle tecnologie meno diffuse (ad esempio, nel presente articolo, gli Scriptlet), sia sperimentare soluzioni applicative alle quali non sono abituati, rivendibili nella propria attività.

In questo primo articolo realizzeremo, in una dozzina di modi differenti, una semplice Tabella Pitagorica. L'argomento è talmente semplice che non è necessaria nessuna preventiva spiegazione. Questo ci consentirà di concentrarci esclusivamente sulle tecniche di programmazione e sulle varie tecnologie.

ed utilizzando un foglio Excel come "destinatario" dei valori. Noi useremo prevalentemente la coppia di istruzioni For .. Next, che, utilizzando una variabile numerica, forniscono gratis il numero di colonna e quello di riga.

Le istruzioni For .. Next trattano direttamente numeri, mentre i cicli basati sulle istruzioni Do .. While e similari si basano sulla valutazione di un'espressione logica, utilizzate anche per creare cicli numerici.

In italiano tradurremo "For K = 1 to 10", in conta da 1 a 10 ed il valore assegnato alla variabile K, ed invece "Do While K<=10" in vai avanti finché è vero che K è inferiore oppure uguale a 10.

Cosa serve per sperimentare i vari esercizi

Realizzeremo i primi esercizi con il Visual Basic, il prodotto Visual Basic, dapprima in modo semplice e poi realizzando un componente, che potremo materializzare in una libreria DLL riutilizzabile in altri ambienti, prima di tutto Internet all'interno di pagine ASP.

Programmeremo in MS Office, sfruttando il Visual Basic for Application, e faremo una puntata in OLE Automation e poi in Windows Hosting Scrip-

ting, che è presente in Windows 98.

Passeremo poi decisamente in Internet per costruire Tabelle Pitagoriche sia totalmente sul Client, sfruttando i linguaggi interpretati lato client dal nostro browser, sia facendo lavorare il server.

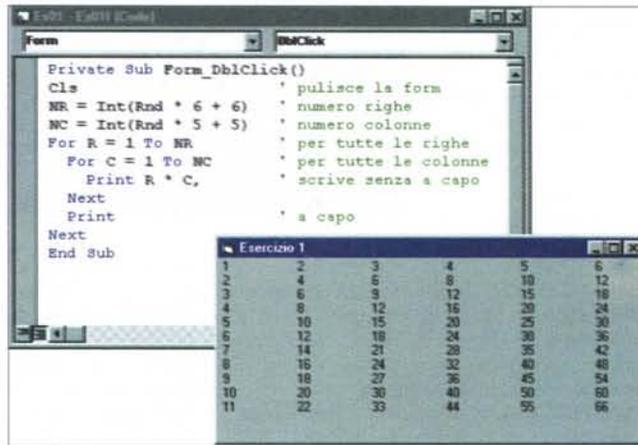
Per provare Visual Basic Script oppure JavaScript inserito in una pagina Internet basta utilizzare un qualsiasi editor e salvare con suffisso HTM oppure HTML e poi richiamare il tutto dal browser che provvede ad interpretare il codice ed a visualizzare il risultato.

Per sperimentare la programmazione delle pagine ASP lato server occorre ovviamente un server. La soluzione

più economica è quella di utilizzare il Personal Web Server che rende un qualsiasi Windows 95 oppure 98 un server Internet in grado di interpretare il codice lato server e di tradurlo in codice HTML. Una pagina ASP può accedere a database di qualsiasi tipo, sfruttando la tecnologia ADO, e può utilizzare DLL, anche quelle fatte da noi.

I primi esercizi in MS Visual Basic

Non commenteremo molto gli esempi, sia perché in pratica ne abbia-



MS Visual Basic e l'istruzione Print. Il primo programma viene eseguito al verificarsi dell'evento "doppio click" sul form. Vengono impostate due variabili, NR e NC, numero di righe e numero di colonne, che assumono valori casuali. Poi con due cicli For .. Next vengono calcolati tutti gli elementi della Tabella Pitagorica che vengono visualizzati con una semplice istruzione Print. Come in tutti i programmi, anche nei più complessi, occorre distinguere i comandi di "calcolo" da quelli che si occupano della "visualizzazione" dei dati.

Umberto Eco, nel suo primo articolo del 2000 apparso all'inizio di gennaio su La Repubblica, parla di Informatica. Parla molto di Internet, ovviamente, ma non trascura assolutamente la programmazione, affermando testualmente che "se si insegna ad un bambino un linguaggio informatico, questo esercizio logico lo renderà padrone e non schiavo del computer".

Quindi, per Eco, lo studio di un linguaggio informatico ha comunque una valenza "formativa" che va ben oltre la sua finalità diretta, che dovrebbe essere la realizzazione di applicazioni informatiche.

Alla base di qualsiasi linguaggio informatico ci sono infatti dei concetti logici assoluti che prescindono dall'informatica, che, e stavolta cito un mio modo di dire, "esistono in natura, indipendentemente dall'esistenza dell'uomo".

Un'Equazione di Secondo Grado esiste in natura, la Matematica Booleana (che fa funzionare il computer) esiste in natura, la relazione Uno a Molti (che fa funzionare i database sui computer) esiste in natura.

MCmicrocomputer inizia una serie di articoli dedicati alla programmazione avendo come finalità non la realizzazione di un'applicazione, ma l'approfondimento di un tema generale, che sarà trattato in modo assolutamente astratto ed il cui studio servirà sia a chi del

A chi serve saper programmare?

computer sarà solo un utilizzatore "normale", sia a chi invece vuole diventare un utilizzatore "esperto", sia a chi vuole diventare un "tecnico informatico".

Argomento del primo articolo è la realizzazione di una Tabella Pitagorica, argomento astrattissimo, che non richiede nessuna conoscenza e nessuna spiegazione, che può essere affrontato in maniera elementare dal punto di vista del programmatore, e che pur nella sua semplicità movimentava alcuni concetti fondamentali. Realizzeremo la nostra Tabella Pitagorica in una dozzina di maniere differenti, usando diversi linguaggi e diverse situazioni ambientali: ambiente Windows, tecnologia COM, ecc. Per quanto riguarda i linguaggi di programmazione, useremo prevalentemente il Visual Basic ed i suoi vari dialetti, con qualche "puntatina" in altri linguaggi, nei casi in cui la tecnologia che si sta usando lo richiedesse: ad esempio, già in questo primo articolo, il JavaScript. Nel prossimo articolo parleremo di date, che, utilizzando un'espressione citata poco fa, "non esistono" in natura, ma rappresentano una convenzione, alquanto astrusa (al limite della compatibilità con il computer, vedi millennium bug) dell'uomo per rappresentare il tempo.

MS Visual Basic ed il componente Grid.

Lo scopo del secondo esercizio, sempre e rigorosamente Visual Basic, è quello di riempire una griglia MSFlexGrid con i valori della Tabella Pitagorica. Mostriamo due "varianti", la prima prevede il preventivo dimensionamento della griglia ed il successivo riempimento delle caselle, la seconda prevede il calcolo degli elementi e la contestuale "crescita" della griglia, ottenuta sfruttando il metodo AddItem.

```
Private Sub P1_Click() ' griglia predimensionata
NR = Int(Rnd * 6 + 6): G1.Rows = NR
NC = Int(Rnd * 5 + 5): G1.Cols = NC
G1.ColWidth(-1) = 400
For R = 0 To NR - 1
  For C = 0 To NC - 1
    G1.Col = C
    G1.Row = R
    G1 = (R + 1) * (C + 1)
  Next
Next
End Sub
Private Sub P2_Click() ' righe aggiunte dinamicamente
NR = Int(Rnd * 6 + 6): G1.Rows = NR
NC = Int(Rnd * 5 + 5): G1.Cols = NC
G1.ColWidth(-1) = 400
For R = 0 To NR - 1
  RG = "" ' riga da aggiungere
  For C = 0 To NC - 1
    RG = RG & (R + 1) * (C + 1) & Chr(9)
  Next
  G1.AddItem RG
Next
End Sub
Private Sub Form_Load()
  G1.FixedCols = 0
  G1.FixedRows = 0
End Sub
```

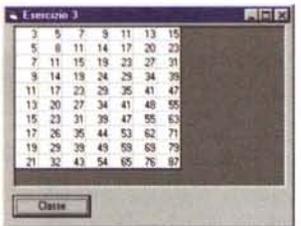


MS Visual Basic ed un componente che restituisce una Tabella Pitagorica.

Proponiamo una variante più tecnica del problema della Tabella Pitagorica. Costruiamo una classe (Es03sp.Es03sc) che contiene una funzione, chiamata Pitg(), che a sua volta restituisce una matrice contenente i dati della Tabella Pitagorica (in gergo C/S, questo è il server). Il relativo listato lo vediamo nella finestra sulla sinistra. Sullo sfondo vediamo come referenziare la classe e come richiamare la funzione dall'applicazione client. In pratica la funzione Pitg restituisce una matrice che viene passata alla variabile MM. Le due funzioni UBound permettono di ricavare le dimensioni della matrice, che non sono note al client in quanto sono state impostate a valori casuali sul server.

```
Dim PT As Es03sp.Es03sc
Private Sub P1_Click()
Set PT = New Es03sp.Es03sc
MM = PT.Pitg
NR = UBound(MM, 1): G1.Rows = NR: G1.FixedRows = 0
NC = UBound(MM, 2): G1.Cols = NC: G1.FixedCols = 0
G1.ColWidth(-1) = 400
For R = 1 To NR
  For C = 1 To NC
    G1.Row = R - 1: G1.Col = C - 1: G1 = MM(R, C)
  Next
Next
End Sub
```

```
Public Function Pitg() As Variant
Dim MM()
NR = Int(Rnd * 6 + 6)
NC = Int(Rnd * 5 + 5)
ReDim MM(1 To NR, 1 To NC)
For R = 1 To NR
  For C = 1 To NC
    MM(R, C) = R * C + R + C
  Next
Next
Pitg = MM
End Function
```



Nel primo esercizio citiamo la presenza dell'istruzione PRINT, che era l'unica istruzione di Output nei Basic di 20 anni fa, che non avevano nulla di visuale.

L'aspetto interessante del secondo esercizio sta nel fatto che mostra i due modi per alimentare una griglia, puntando via via le celle, se la griglia è stata già dimensionata, oppure sfruttando il metodo Additem, che invece crea via via righe nuove (righe e non colonne), che vengono raggiunte grazie al carattere 9, che corrisponde al tabulatore.

Il terzo esercizio usa una classe in cui abbiamo creato una Function Pitg() che restituisce una matrice contenente i numeri della Tabella Pitagorica. Compito dell'applicazione è quello di visualizzare il contenuto della matrice, che a questo punto potrebbe contenere qualsiasi cosa, nella griglia.

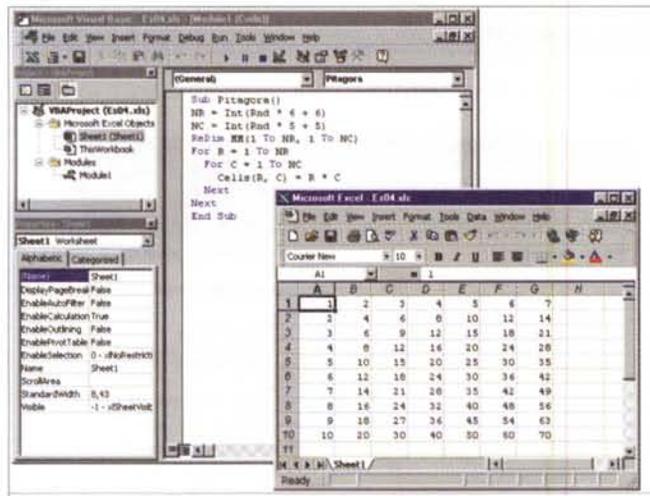
Per chi usa MS Office

Il linguaggio di programmazione di MS Office è il VBA, ed è analogo in termini di istruzioni e di funzioni al fratello maggiore VB. In più il VBA permette di manipolare gli oggetti del prodotto da cui è ospitato.

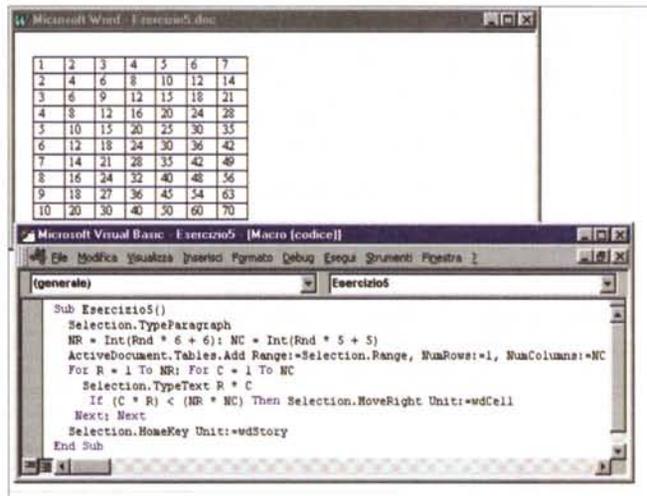
Nel quarto esercizio vediamo come si usa l'oggetto Cell per scrivere, ma si può anche leggere, sul foglio di Ex-

mo già parlato nell'introduzione, sia perché le didascalie sono molto esplicative, sia perché abbiamo preferito

proporre più esercizi e meno... chiacchiere, a parità di spazio a noi riservato.



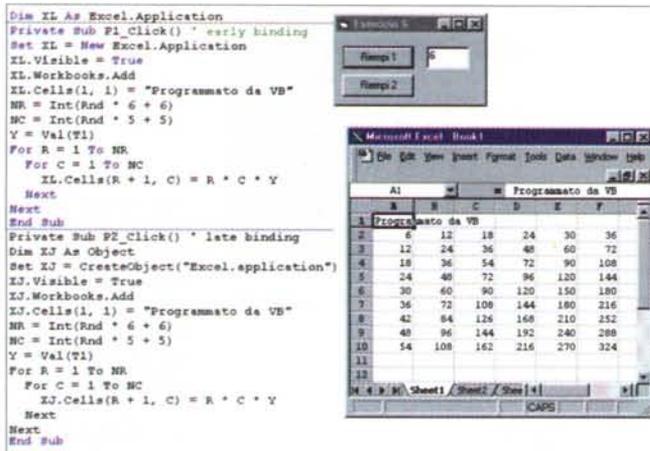
Un foglio di MS Excel riempito con VBA. I successivi tre esercizi sono realizzati con Excel, Word ed ancora Excel. Il primo, qui rappresentato, consiste in una Macro Excel che esegue il solito calcolo dei dati della Tabella Pitagorica. In questo caso per riempire un foglio di Excel viene utilizzato l'oggetto Cells(r,c) che permette di dialogare, sia in lettura che in scrittura, con il foglio stesso.



Un documento MS Word riempito con VBA. Con Word il discorso è lo stesso solo che, non essendo disponibile il foglio a quadretti di Excel, va creato, nel foglio bianco di Word, una tabella. Si usa l'oggetto "ActiveDocument". Della tabella basta dimensionare il numero delle colonne, in quanto con il comando di spostamento Selection.MoveRight vengono via via aggiunte le celle che servono.

Un foglio di MS Excel prodotto da VB sfruttando OLE Automation.

La tecnologia OLE Automation permette di vedere le applicazioni Office come componenti OLE programmabili. Nell'esercizio che vi proponiamo realizziamo la Tabellina Pitagorica in un programma Visual Basic (che ha un semplice form con due pulsanti) che crea una istanza di Excel e la riempie con i soliti dati. Le due varianti, corrispondenti ai due pulsanti, propongono i due modi di istanziare un oggetto Excel. L'Early Binding, che passa per la dichiarazione di un oggetto Excel, e nella sua successiva istanziazione, e il Late Binding, che prevede la dichiarazione di un oggetto generico, che solo al momento della istanziazione diventa un oggetto Excel. La casellina nel form VB contiene un numero da usare come moltiplicativo di ciascun elemento della cella.



cel. Nel quinto creiamo una tabella usando il VBA di Word.

E' evidente che buona parte del codice che serve si può generare usando il registratore di macro per poi adattarlo alle necessità dell'applicazione. In questo caso è stata generata una tabella di una sola riga che, tabulando tabulando, viene ingrandita.

Nella figura 6 un esercizio di OLE Automation. Un'applicazione VB che esegue il calcolo della Tabella Pitagorica che viene "sparata" su un foglio Excel. La tecnologia OLE Automation prevede l'esistenza di un server (nel nostro caso Excel) che viene manipolato da un client.

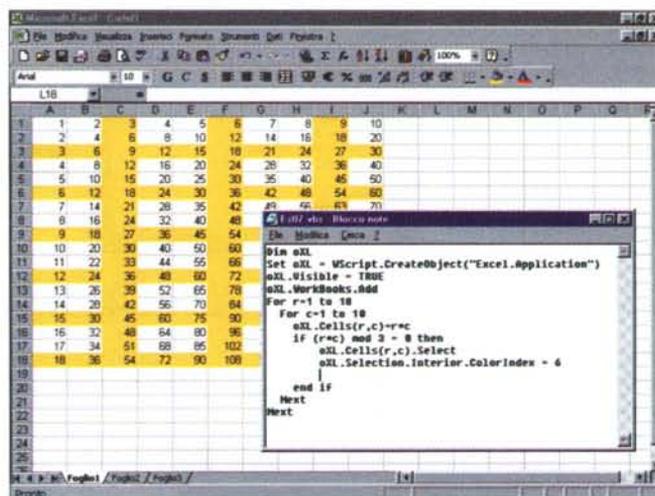
La figura mostra un piccolo form di Visual Basic in cui viene scritto un numero che poi sarà usato come moltiplicatore della Tabella Pitagorica. Abbiamo anche usato i due modi con i quali si può referenziare un'applicazione esterna.

Una variante di OLE Automation è Remote Automation, che prevede che il server risieda fisicamente su un'altra macchina. La tecnologia Remote Automation precorre la tecnologia DCOM, che poi, in Windows 2000, si associa al vecchio MS Transaction Server e confluisce nel COM+. MTs+DCOM=COM+.

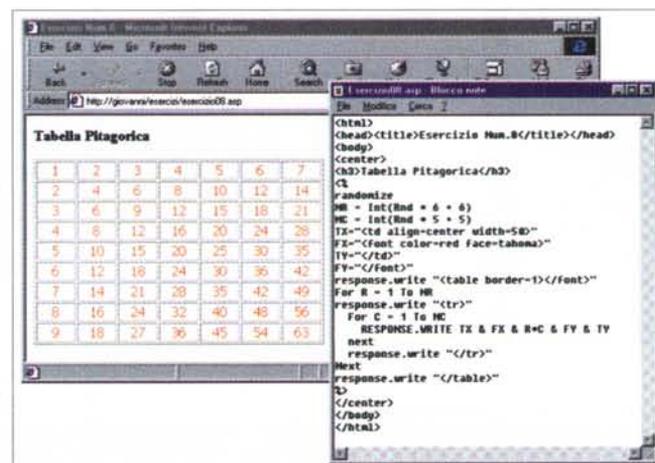
Cosa è il Visual Basic Script

Il settimo esercizio è una variante del precedente, solo che l'ospite, ovvero il programma che lancia Excel e

Tabella Pitagorica realizzata con Windows Hosting Script verso Excel. Si può praticare OLE Automation partendo da un qualsiasi ambiente programmabile in VB o VBA. In Windows NT 4.0, Windows 2000 e Windows 98 è disponibile la tecnologia Hosting Script che consente di eseguire codice Visual Basic Script (e anche JavaScript) direttamente dalla riga di comando di Windows. Trattandosi di Script si può usare, per referenziare l'oggetto Excel, solo il Late Binding. Nella figura vediamo il codice scritto con un comune notepad e salvato con suffisso VBS. A tale suffisso è associato il programma WScript.exe che si incarica di interpretare e quindi di eseguire il codice.



Pagina Active Server Pages (ASP) - Una Tabella Pitagorica prodotta da un server Internet. I successivi tre esercizi prevedono il browser Internet come ambiente in cui visualizzare la nostra Tabella Pitagorica. In questo primo esercizio il codice ASP viene eseguito dal server che produce un codice HTML che viene inviato al client. Per poterlo sperimentare occorre disporre di un server IIS (MS Internet Information Server) oppure del PWS (Personal Web Server, di cui parliamo nel testo). Nel secondo e nel terzo caso sfrutteremo una programmazione lato client. Sarà quindi il browser stesso che si incaricherà di interpretare il codice, scritto in Visual Basic nell'esercizio 9 ed in JavaScript nell'esercizio 10, per produrre la tabella.



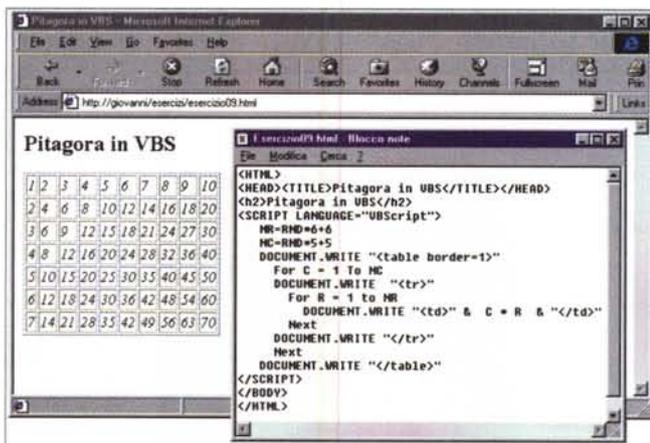
ci scrive dentro, non è il Visual Basic ma è lo stesso Windows. E' Windows 98 che dispone dell'Hosting Script.

In pratica si può scrivere, con un editor qualsiasi, del codice Visual Basic Script, che in quanto interpretato ha delle limitazioni rispetto al capostipite e poi si può lanciare direttamente con un semplice doppio click sul file, che deve però avere desinenza VBS.

E' evidente che a questo tipo di file, ma anche a quelli di tipo JS, è associato un programma lanciatore, che si chiama WSCRIPT.EXE e che si incarica di interpretare ed eseguire il codice. L'Hosting Script è disponibile anche in Windows NT ed in Windows 2000.

Tanto per vivacizzare il risultato abbiamo colorato di giallo le celle il cui contenuto è un numero divisibile per 3.

Pagina HTML - Tabella Pitagorica prodotta dal client in VBScript. Da notare come sia l'esercizio ASP che quello VB utilizzino i soliti due cicli For .. Next. Le istruzioni di visualizzazioni inserite all'interno del codice sono Response.Write nel caso di ASP e Document.Write nel caso di VB. Tali due istruzioni sono quelle che generano il codice HTML, che poi viene visualizzato sulla pagina. Vengono quindi predisposte delle stringhe che assemblano sia i dati da visualizzare che i codici HTML necessari per la creazione della tabella.



quando facciamo click sul pulsante chiamato Pitagora.

Lavorare con i componenti

Gli ultimi esperimenti li conduciamo su una tecnologia complessa come funzionamento, ma semplicissima da mettere in pratica (perché la si può mettere in pratica senza necessariamente conoscerne il meccanismo di funzionamento interno): la tecnologia dei componenti.

Si tratta di scomporre l'applicazione in due o più pezzetti, ognuno dei quali

Internet lato server e lato client

Vogliamo utilizzare Internet per produrre e per visualizzare una Tabella Pitagorica.

La prima decisione da prendere è se far lavorare il client oppure il server. Nel caso in cui optiamo per il server si può utilizzare la tecnologia ASP, presente in Internet Information Server, che è una parte di Windows NT, oppure nel Personal Web Server, ambedue della Microsoft.

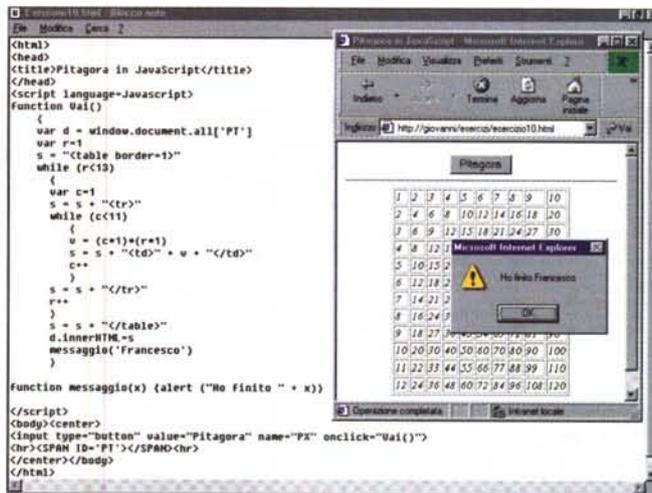
Dal punto di vista operativo occorre creare un file con desinenza ASP in cui oltre ad un normale codice HTML possiamo inserire codice VBScript o JavaScript, identificabile dallo speciale TAG "<% .. %>".

Nel caso venga richiamato, tale file è il "motore" del server che scorre il file ASP, individua le porzioni di codice e le esegue. Il risultato di questa esecuzione viene inviato al client, che non è poi in grado di "vedere" il codice ma solo il risultato della sua esecuzione.

Nel caso della soluzione lato client, il server trasmette codice VBScript o JavaScript, codice che quindi può essere visto dal client, ed è il browser che si incarica di interpretarlo e di mostrarlo sulla pagina.

Negli esercizi che vanno dal numero 8 al numero 10, vediamo prima la soluzione ASP, quindi server, e poi le due soluzioni client, la prima VBS e la seconda JS.

In altri casi, anzi in buona parte degli altri casi, non è indifferente il fatto che lavori il server o il client. Ad esempio se la pagina prevede l'acces-



mico, in cui vengono identificati e manipolati gli elementi della pagina.

Pagina HTML - Tabella Pitagorica prodotta dal client in JavaScript

Il linguaggio JavaScript è più simile al C++ che non al Visual Basic. Vediamo i due cicli While, che iniziano e finiscono con le parentesi graffe, vediamo le istruzioni che servono per incrementare i valori di R e di C. In questo caso, più elaborato rispetto a quello realizzato in Visual Basic, la Tabella Pitagorica viene prodotta al click sul bottone PX piazzato in alto nella pagina. Viene creata una stringa S che contiene il codice HTML con la tabella e che viene "sparata" nella posizione definita dal tag SPAN. Questo è un esempio di HTML dina-

so ad un DB che sta sul server la tecnologia è ASP, se invece occorre eseguire dei controlli formali sui dati digitati in un form, la tecnologia è client, si usa o JavaScript (interpretato, con alcune differenze di comportamento, da MS Internet Explorer e da Netscape Navigator) o Visual Basic Script (interpretato solo da Explorer).

Nell'esercizio 10 ci siamo spinti ad utilizzare un po' di DHTML. In pratica abbiamo individuato un elemento della pagina sfruttando il Tag "" e chiamandolo PT. Poi con una funzione JavaScript abbiamo creato un codice HTML che produce una Tabella Pitagorica. Infine abbiamo associato all'evento click su un pulsante inserito nella pagina il passaggio di tale codice all'elemento PT.

In pratica la Tabella Pitagorica, pre-ceduta da un messaggio, appare

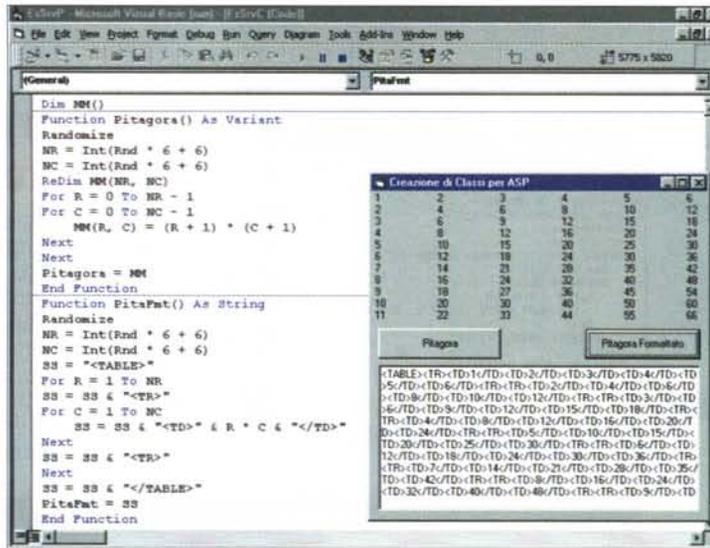
svolge una certa funzione e ognuno dei quali comunica con l'esterno tramite la cosiddetta Interfaccia.

Il primo esperimento lo facciamo con il Visual Basic, con il quale realizziamo un ActiveX di tipo DLL (EsSrvP), che contiene un modulo di classe (EsSrvC), che a sua volta contiene due funzioni Pitagora() e PitaFmt(). La prima produce una matrice numerica con i dati della Tabella Pitagorica. La seconda produce una stringa che contiene direttamente il codice HTML che produce la Tabella Pitagorica. Si tratta dunque di due funzioni semplicissime che non hanno parametri in entrata e ne hanno uno solo in uscita.

E' evidente che la seconda funzione, quella che produce una stringa con codice HTML, può essere utilizzata solo all'interno di una pagina ASP. Questa soluzione contrasta con la teo-

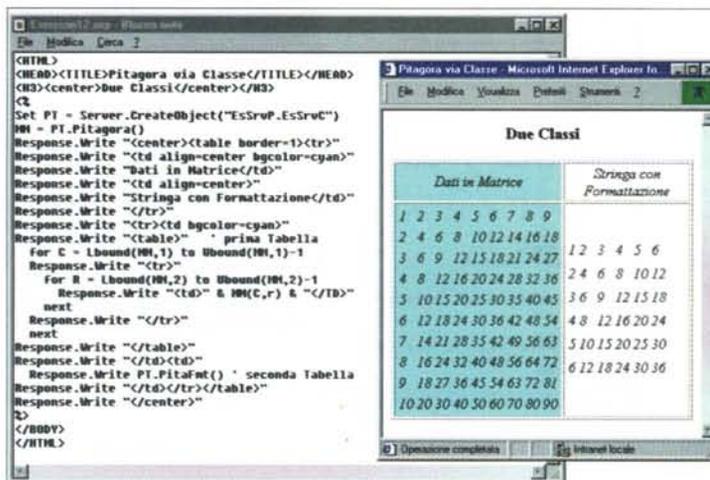
MS Visual Basic - Creazione di due funzioni da usare in pagine ASP.

Questo esercizio prevede la realizzazione, eseguita con MS Visual Basic, di un ActiveX di tipo DLL, che comprende due funzioni, Pitagora(), che restituisce una matrice contenente i dati della Tabella Pitagorica, e Pitafmt(), che restituisce una stringa con il codice HTML necessario per visualizzare nel nostro browser la Tabella Pitagorica. E' evidente che questa seconda funzione è utilizzabile solo in un'applicazione ASP.



Uso, in una pagina ASP, della classe con le funzioni Pitagora() e Pitafmt().

Ecco, in una sorta di "slalom parallelo", le nostre due funzioni inserite in una pagina ASP. La seconda, che prepara direttamente codice HTML, va semplicemente richiamata, mentre la prima, che restituisce una matrice con i dati, necessita di un ulteriore lavoro di impaginazione.



Uso, in una pagina HTML, di uno Scriptlet JavaScript.

Un'ennesima variante applicativa del nostro problema è costituita dallo Scriptlet. Si tratta di una sorta di componente, nel nostro caso è stato realizzato in JavaScript, contenente delle funzioni, definibili e richiamabili come object da un programma chiamante. Qui vediamo una funzione Pitagora, che restituisce una stringa, ed una funzione Saluto. Attenzione: abbiamo inserito nella stessa videata i listati dei due file, in alto il chiamante, ed in basso il chiamato, lo Scriptlet.



ria dei componenti che prevedono, anzi pretendono, che i componenti siano neutri e quindi riutilizzabili nelle più svariate situazioni.

Nella figura 11 vediamo il codice delle due funzioni ed un form Visual Basic che le utilizza. Il risultato della prima lo visualizziamo con la solita Print, mentre il risultato della seconda lo vediamo in una TextBox, che non è in grado di formattare il codice HTML (potevano usare un componente IE che visualizza il codice HTML, ma ci saremmo troppo allontanati da... Pitagora).

Nella figura 12 una pagina ASP che mostra affiancati i risultati prodotti dalle due funzioni. La prima produce una matrice che scarichiamo in una table, la seconda produce un codice HTML già completo e lo dobbiamo solo richiamare con due, due, righe di codice:

```

Set PT = Server.CreateObject("EsSrvP.EsSrvC")
Response.Write PT.Pitafmt()
    
```

Anche nella programmazione per Internet è possibile utilizzare componenti scritti con i linguaggi propri del mondo Internet. Un componente, richiamabile come object da un'applicazione client, e che mostra un'interfaccia, fatta di proprietà e metodi, che si chiama Scriptlet.

Come realizzare la nostra Tabella Pitagorica sotto forma di Scriptlet lo vedete nell'ultima figura, la 13.

Conclusioni

Nel prossimo numero vorrei parlare di date, date osservate dal punto di vista del programmatore.

Ormai siamo nel 2000 e fortunatamente ci siamo lasciati alle spalle il problema del millennium bug, che quindi non tratteremo.

Vedremo dapprima come "funzionano" le date, che diciamo subito, non sono né numeri né tanto meno stringhe, e poi cercheremo di padroneggiarle nei vari casi, e saranno frequentissimi, in cui ci troveremo a doverle maneggiare.

internet **no** **limit**

la nuova generazione di abbonamenti ADSL di MC-link per le Aziende

Dopo la rivoluzione di Internet ecco la rivoluzione della nuova generazione di abbonamenti aziendali MC-link, Internet no limit: ADSL Lan + e ADSL Corporate. Grazie alle nuove tecnologie ADSL e ATM, basta attese per collegarsi, basta attese per ricevere o inviare file pesanti, basta scatti telefonici, basta linea occupata mentre siete su Internet... basta un clic e siete subito dove volete, senza limiti. Perché scegliendo Internet no limit di MC-link la vostra azienda è sempre on-line, connessa con il resto del mondo 24 ore su 24 senza ulteriori costi telefonici. E naturalmente mai soli. A tenervi compagnia ci sono i nostri servizi, la nostra assistenza e i nostri 25.000 clienti che hanno già scelto la qualità Internet di MC-link.

R.C.P. associati

	ADSL LAN+ consigliato per LAN Aziendali fino ad 8 postazioni	ADSL Corporate consigliato per LAN Aziendali con oltre 10 postazioni
Disponibilità del servizio al 1/2/2000 (altre sedi in via di attivazione)	Roma e Milano	Roma, Milano, Genova, Torino, Napoli, Firenze e Bologna
Velocità di connessione (ricezione/trasmisione)	fino a 640/128K	fino a 640/128K
Indirizzi IP statici per la connessione di Webcam e Server Internet (Web, FTP, Mail, etc.)	1	14
Consigliabile per videoconferenza	no	si
Domini (.it, .com, .org o .net)	1	3
Caselle di posta su Mail Server personalizzato con il dominio aziendale residente c/o MC-link e configurabile on-line	10	25
Spazio Web Commerciale con il dominio aziendale (www.azienda.it) e comprensivo di tool per commercio elettronico	10MB	50MB
Account dial-up aggiuntivi per accessi da 4.000 comuni italiani e 1.200 città nel mondo	10	25
Consultazione gratuita della Gazzetta Ufficiale (normativa nazionale, regionale e comunitaria)	si	si
Canone mensile	lit. 240.000 + Iva	lit. 750.000 + Iva
Costi di attivazione	lit. 400.000 + Iva	lit. 400.000 + Iva

Fino al 30/4/2000 risparmia oltre un milione grazie al router* ADSL incluso.