

Lepri e tartarughe

Questo mese intendo presentare i risultati di un paio di anni di misurazioni della performance di Mathematica su diversi elaboratori, soprattutto Mac ma anche Sun, Next, HP e PC IBM-compatibili, con diverse combinazioni di sistemi operativi. Oltre che interessanti di per sé questi dati permettono una discussione più generale delle tecniche empiriche di test sui personal computer e sulla loro affidabilità

di Francesco Romani

Introduzione

Da quando sono entrati in funzione i primi elaboratori si è innescata una corsa frenetica e inarrestabile a migliorarne le prestazioni sia aumentando la potenza dell'hardware sia razionalizzando il software. Purtroppo potenza e semplicità non sono mai andate d'accordo. Lo sviluppo di sempre maggiori potenzialità e la realizzazione di interfacce sempre più raffinate (es. l'interfaccia Mac, Windows, ecc.) porterebbe ad un inevitabile catastrofico degrado delle prestazioni, se contemporaneamente, l'hardware non aumentasse la sua potenza di svariati ordini di grandezza. I sonni di ogni utente abbastanza evoluto (e un po' maniaco come il sottoscritto) sono tormentati da dubbi del tipo: mi conviene installare la versione 7.0.1.b-bis? Aggiungo questa manciata di init per avere le icone gialle a pallini rossi? Piuttosto mi conviene semplificare il sistema e lavorare in bianco e nero per migliorare le prestazioni? Come posso verificare quale è il compromesso migliore?

Da questo punto di vista *Mathematica* rappresenta un caso emblematico. Le prestazioni straordinarie del sistema in termini di versatilità e potenza si pagano con una, altrettanto straordinaria, fame di risorse hardware (memoria RAM e velocità di CPU) e software (sensibilità ai bug del sistema). In questa pagina presento i risultati di un programma di benchmark messo a punto per sperimentare le prestazioni di *Mathematica* in varie configurazioni e su tutte le macchine significative che sono riuscito a raggiungere. Per alcuni problemi per cui ciò era facilmente realizzabile ho anche fatto un confronto con l'efficienza di programmi Pascal equivalenti, fatti girare su una stessa macchina.

Descrizione del test generale

L'idea di base è provare un insieme significativo di funzioni di *Mathematica* raccolte in un modulo facile da eseguire e di durata tale da permettere l'esecuzione anche su macchine disponibili per pochi minuti (avete mai provato a chiedere se vi fanno usare per un paio d'ore quella macchina ultimo modello in esposizione?). Il modulo completo è riportato nell'incorniciato in modo che il lettore volenteroso può fare lo stesso test sulla sua macchina e confrontare i risultati. Vediamo ora passo passo cosa viene calcolato.

Dopo avere stampato alcune informazioni di riferimento (la data del test, la versione di *Mathematica* usata e il sistema su cui avviene l'elaborazione) vengono fatte varie elaborazioni, elencate nel seguito, con una sigla che servirà per i riferimenti successivi. Gli asterischi stanno a significare che la prova in questione non è implementabile in un linguaggio tradizionale (C, Pascal, Fortran, Basic) senza un notevole sforzo di programmazione o l'uso di speciali librerie.

• PLOT, un plottaggio bidimensionale di due funzioni oscillanti.

• NINT1, un'integrazione numerica in precisione di macchina.

• NINT2, la stessa integrazione numerica, ripetuta una seconda volta. La seconda elaborazione si avvantaggia delle tabelle che il programma *NIntegrate* si è calcolato la prima volta.

• NINT3*, come NINT1 ma lavorando con 50 cifre decimali e richiedendo 30 cifre decimali di precisione nel risultato.

• NINT4*, NINT3 ripetuta.

• INVS*, inversione simbolica di una matrice 4x4 (per la cronaca è la matrice $\{\sin(\pi i j/9)\}_{ij}$).

• INV, inversione numerica, in precisione di macchina, di una matrice 30x30.

• INVMP*, inversione numerica, lavorando con 100 cifre decimali di precisione di macchina, di una matrice 8x8.

• INVR*, inversione esatta di una matrice 15x15 di numeri razionali.

• SER*, il calcolo simbolico dello sviluppo in serie con 15 termini di $f(x) = \cos(\pi x^2+1)/\cos(x-1)$, (provate un po' a farlo a mano o in C!!).

• INTS1*, sviluppo in serie di $f(x)$ con 2 termini e integrazione simbolica del polinomio risultante.

• INTS2*, ripetizione della integrazione per avvantaggiarsi dei lavori di inizializzazione.

• LOOP, tre *Do* annidati per eseguire 8000 volte l'istruzione nulla.

Il programma stampa per ogni test il tempo parziale e il tempo accumulato e viene chiamato con il comando

```
Print["TotalTime =",
      Timing[test][[1]]]
```

che stampa anche il tempo totale.

Questo test non ha alcuna intenzione di essere definitivo o completo o ben scritto, probabilmente scrivendolo oggi lo farei diverso e più completo, ma per assicurarne la validità attraverso il tempo, ho preferito lasciarlo com'era.

Le macchine usate per le prove sono le seguenti

- 170: Mac Powerbook 170, 68030, 25 MHz, sistema 7.0.1
- 180: Mac Powerbook 180, 68030, 33 MHz, sistema 7.1
- 486/33: PC con 486 DX a 33MHz, 4Mb Ram.
- 610: Mac Centris 610, 68LC040 (senza coprocessore), 20 MHz, sistema 7.1
- 650: Mac Centris 650, 68040, 25 MHz, sistema 7.1
- 660: Mac Centris 660 av, 68040, 25 MHz, sistema 7.1
- 700: Mac Quadra 700, 68040, 25 MHz, sistema 7.0.1
- 800: Mac Quadra 800, 68040, 33 MHz, sistema 7.1
- 840: Mac Quadra 840 AV, 68040, 40 MHz, sistema 7.1
- Cl: Mac Cl, 68030 25 MHz, sistema 7.0.1
- FX: Mac FX, 68030 40 MHz, sistema 7.0.1
- HP: HP Apollo 715 a 25MHz
- NeXT: NeXTStation, 68040, 33 MHz.

- PEN: OLIDATA con Pentium Intel a 66MHz, 8Mb Ram.
- SUN: Sun SparcStation 4, 25 MHz.

Prima di passare alla presentazione dei risultati alcune considerazioni carattere generale.

Prima di tutto questi test non misurano l'efficienza della CPU del sistema in prova, semplicemente perché ciò non è possibile! Ogni test di qualunque tipo produce una media pesata delle performance della macchina su diversi problemi e il risultato è influenzato da moltissimi fattori: la velocità del disco, la presenza di memoria interallacciata, la velocità della scheda video, la presenza di un coprocessore matematico, la larghezza del bus di sistema, etc. I test sono stati fatti sulle macchine "in ordine di marcia" così come erano disponibili durante il normale funzionamento. Ciò implica che molti fattori esterni possono perturbare il risultato del test. Ad esempio la Sun è utilizzata per alcune attività di esercitazione degli studenti del corso di Laurea in Scienze dell'Informazione e, a seconda dei momenti, può essere molto carica o completamente scarica. In ambiente Mac particolari INIT (p.es. Super-Clock) rallentano l'esecuzione, la visualizzazione con molti colori o livelli di grigio è più lenta di quella in bianco e nero, il collegamento in rete richiede un certo numero di attività, trasparenti per l'utente, ma costose per il sistema, ecc. Sotto Windows la presenza o meno delle operazioni di swap e la qualità della scheda video influenzano grandemente le prestazioni.

Per entrare nella polemica sollevata su un recente numero

di MC, si può affermare che certamente i computer non hanno un'anima e sono costruiti da ingegneri e non da poeti, ma tante sono le variabili in gioco e soprattutto tante di esse sono così abilmente nascoste che, a tutti gli effetti, è come se un'anima ce l'avessero (e in base alle note leggi di Murphy pure molto dispettosa). Per filtrare un po' di queste variabili aleatorie di solito si incrociano altri test, in condizioni diverse per avere un'idea dell'influenza dei vari parametri, ottenendo una serie di risultati certamente indicativi. Comunque, pretendere di misurare in assoluto l'efficienza di una macchina o di una implementazione di un linguaggio con un numero finito di test, sarebbe perlomeno ingenuo.

Risultati

Vediamo una prime serie di risultati per le varie porzioni intermedie del test effettuati con la versione 2.2 (l'ultima in distribuzione alla fine di settembre '93).

I lettori più accorti noteranno che, nella prima serie di risultati, la somma dei tempi parziali differisce dal tempo totale, ciò è corretto in quanto quest'ultimo tiene conto anche del sovraccarico del programma `test`, (soprattutto le istruzioni di stampa). Un altro fatto strano è che per la Sun il tempo della seconda integrazione simbolica è superiore a quello della prima. Si possono fare varie ipotesi: prima di tutto è una legge generale che i benchmark non danno mai due volte lo stesso risultato (sempre in base alle considerazioni, di cui sopra,

Il programma di benchmark

```
(* benchmark per Mathematica, by Francesco Romani, 1993.
è permesso copiarlo e distribuirlo ma non modificato,
per ovvi motivi di coerenza nei risultati *)
test:=Module[{t=0,w,w1,tp,tp1,tp2,i,j,k,h},
Print[" Begin Test          ",Date[ ]];
Print[" System              ",$System];
Print[" Mathematica version  ",$Version];
Print["                      t = 0"];
Print[" Plot "];
Print[tp=Timing[Plot[{x
Cos[x^2],Sin[x^2]},{x,0,10}][[1]]];
Print["                      t =",t,t=+tp];
Print[" "];
Print[" NIntegrate, machine precision"];
Print[tp1=Timing[NIntegrate[x Cos[x^2]},{x,0,10}][[1]]];
Print[tp2=Timing[NIntegrate[x Cos[x^2]},{x,0,10}][[1]]];
Print["                      t =",t,t=+tp1+tp2];
Print[" "];
Print[" NIntegrate, 50 digits"];
Print[tp1=Timing[NIntegrate[x Cos[x^2]},{x,0,2},
PrecisionGoal->30,
AccuracyGoal->30,
WorkingPrecision->50][[1]]];
Print[tp2=Timing[NIntegrate[x Cos[x^2]},{x,0,2},
PrecisionGoal->30,
AccuracyGoal->30,
WorkingPrecision->50][[1]]];
Print["                      t =",t,t=+tp1+tp2];
Print[" "];
Print[" Symbolic 4 x 4 matrix Inversion "];
Print[tp=Timing[w=Table[Sin[Pi(i j)/9]},{i,4},{j,4}];
w1=Inverse[w][[1]]];
Print["                      t =",t,t=+tp];
Print[" "];
Print[" machine precision 30 x 30 matrix inversion "];
Print[tp=Timing[w=Table[Sin[Pi(i j)/31]},{i,30},{j,30}];
w1=Inverse[N[w]][[1]]];
Print["                      t =",t,t=+tp];
Print[" "];
Print[" 100 digits 8 x 8 matrix inversion "];
Print[tp=Timing[w=Table[Sin[Pi(i j)/9]},{i,8},{j,8}];
w1=Inverse[N[w,100]][[1]]];
Print["                      t =",t,t=+tp];
Print[" "];
Print[" rational 15 x 15 matrix inversion "];
Print[tp=Timing[w=Table[1/(i+j+1)},{i,15},{j,15}];
w1=Inverse[w][[1]]];
Print["                      t =",t,t=+tp];
Print[" "];
Print[" symbolic computation "];
Print[tp=Timing[
w=Normal[Series[Cos[Pi x^2 +1]/Cos[x-
1]},{x,0,15}][[1]]];
Print["                      t =",t,t=+tp];
Print[" symbolic integration "];
Print[tp1=Timing[
w=Normal[Series[Cos[Pi x^2 +1]/Cos[x-1]},{x,0,2}];
w1=Integrate[w,x][[1]]];
Print[tp2=Timing[w1=Integrate[w,x][[1]]];
Print["                      t =",t,t=+tp1+tp2];
Print[" Looping 8000 times"];
Print[tp=Timing[
Do[Do[h=5,{i,20}]{j,20}]{k,20}][[1]]];
Print["                      t =",t,t=+tp];];
(* una volta definita la funzione test eseguire la riga se-
guente *)
Print["TotalTime = ",Timing[testing][[1]]]
```

Prima serie di risultati

	FX	650	800	SUN
PLOT	4.65	2.6	1.9	1.08
NINT1	6.68	4.21	3.11	2.03
NINT2	6.36	3.95	2.91	1.98
NINT3	31.78	22.00	16.51	6.33
NINT4	18.66	12.83	9.68	3.90
INVS	5.78	3.63	2.76	1.46
INV	7.36	4.93	3.68	2.46
INVMP	9.61	6.43	4.83	2.43
INVR	5.50	3.73	2.76	1.58
SER	24.50	14.73	11.11	6.91
INTS1	9.53	5.83	4.46	1.93
INTS2	8.46	4.98	3.71	2.08
LOOP	7.53	4.50	3.30	2.66
Totale	147.46	95.31	71.48	37.08

sull'anima), un'altra considerazione specifica è che la Sun lavora in Time Sharing per molti utenti e usa massicciamente la memoria virtuale, in ogni momento del test il caricamento di un gruppo di pagine di memoria può perturbare il risultato. Per rendere maggiormente affidabili i test sarebbe stato opportuno fare la media su molte prove e far durare il test molto di più, questo avrebbe però comportato l'impossibilità di fare prove sulle macchine altrui lasciate a disposizione per un quarto d'ora.

Vediamo nella tabella A i risultati del tempo globale per tutti i test effettuati negli ultimi due anni. Nelle prima colonna è riportato il tipo di macchina, nella seconda la versione di *Mathematica* usata (la S sta per versione Standard senza uso del coprocessore) nella terza il sistema operativo nella quarta il tempo totale in secondi, nella quinta le note.

Ed ecco ora i commenti su questi risultati.

Il netto peggioramento da 1) a 2) è dato dalla mancanza della VRAM separata nel Ci con video built-in con conseguente rallentamento degli accessi al primo banco di memoria.

Il netto peggioramento da 5,6) a 7) è dato dalla mancanza del pacchetto Ω -SANE nel sistema 7.1 con conseguente ral-

1)	CI	2.0	7.0.1	179.06	
2)	CI	2.0	7.0.1	243.66	BI
3)	CI	2.1	7.1	234.96	
4)	CI	2.1	7.0.1	236.71	
5)	CI	2.1S	7.0.1	346.91	
6)	CI	2.2S	7.0.1	361.91	FS
7)	CI	2.1S	7.1	535.58	
8)	700	2.0	7.0.1	69.31	
9)	FX	2.0	7.0.1	118.83	
10)	FX	2.2	7.0.1	147.45	
11)	FX	2.2	7.0.1	139.36	NI
12)	SUN	2.0	Unix	35.23 (~50)	
13)	SUN	2.2	Unix	37.08 (~50)	
14)	NeXT	2.1	Unix	~25	
15)	610	2.2S	7.1	272.51	FS
16)	650	2.2	7.1	95.31	
17)	650	2.2S	7.1	135.00	
18)	650	2.2	7.1	93.26	B/N
19)	650	2.2	7.1	90.63	NI
19)	660	2.2	7.1	92.26	NI
20)	800	2.2	7.1	71.48	
22)	170	2.2	7.0.1	321.15	VM
23)	180	2.2	7.1	221.88	NI
24)	840	2.2	7.1	66.66	NI
25)	483/33	2.1	wind.	84.8	
26)	PEN	2.2	wind.	27.62	
26)	HP	2.2	Unix	15.02	

Legenda delle note.

- BI Mac Ci con scheda video built-in
- NI Mac con estensioni disabilitate
- B/N Mac con settaggio in bianco/nero
- VM 4M di memoria reale, 8M di memoria virtuale
- FS File Sharing On

Tabella A

I Mathematica Days 1993

Si sono svolte, alla fine di ottobre, a Parigi, Londra e Francoforte, tre giornate denominate "Mathematica days 1993", per facilitare un incontro tra gli sviluppatori di *Mathematica* e gli utenti e i distributori europei. Ho avuto l'occasione di partecipare alla sessione parigina. Punto centrale della giornata è stata la conferenza del presidente e fondatore della Wolfram, Stephen Wolfram. Sono stati presentati gli sviluppi futuri del sistema insieme con la filosofia che sta alla base del progetto di *Mathematica*: "far fare al calcolatore non quello che l'uomo non sa fare ma tutto quello che il calcolatore può fare (che nell'opinione di Wolfram è evidentemente

molto di più)". Il laboratorio didattico dell'ESIEE presentava una imponente batteria di workstation HP Apollo su cui faceva bella mostra di sé la nuova versione del *FrontEnd* per X-window elegantissima e facile da usare specialmente su schermi da 19" a colori ad altissima risoluzione. (N.B. tale versione del frontend viene ora distribuita gratuitamente della Wolfram agli utenti registrati di *Mathematica* per Unix che hanno sottoscritto il contratto Math-Plus) Nelle aule delle conferenze venivano usati esclusivamente dei Quadra 800, che, ormai scomparse le macchine Next, rappresentano probabilmente la più economica delle piattaforme

solide su cui far girare *Mathematica*. per finire riporto la promessa, da parte di Conrad Wolfram, (fratello di Stephen e General Manager della società) della sollecita messa in commercio di una versione di *Mathematica* compilata per il Power-PC, che, unita alle promesse della Apple di facili upgrade a Power-PC delle più recenti macchine Macintosh, fa sperare nella disponibilità a breve di una piattaforma economica ad altissime prestazioni che permetta al sistema di esplicitare veramente le sue potenzialità. Io sto facendo una corte spudorata ai miei contatti tra i distributori Apple e Wolfram e spero di potervi dare qualche dato prima possibile.

lentamento nell'uso della FPU da parte di programmi che non la accedono direttamente (come le versioni standard di *Mathematica*, scritte per elaboratori senza coprocessore).

Il netto peggioramento da 1) a 3,4), da 9) a 10), da 8) a 16), e da 8) a 20), tenendo conto della differenza dei Clock, è dato da un cambiamento nei programmi interni da *Mathematica*. Tale effetto è evidente in modo diverso da macchina a macchina. Ho posto il problema direttamente a Stephen Wolfram e Jerry Keiper (alla conferenza di Rotterdam del 1992) e mi è sembrato di capire che è un effetto collaterale di importanti ristrutturazioni interne che portano maggiori benefici in altri parametri di misurazione dell'efficienza e, forse, in altri benchmark strutturati diversamente.

Nelle prove sulla SUN, (12) e 13)) il tempo tra parentesi è stato misurato con il cronometro e tiene conto del sovraccarico di sistema, tempi di trasmissione, etc.

La mancanza del coprocessore penalizza fortemente il Centris 610, ottima macchina, dal prezzo strabiliante, ma sconsigliabile a chi deve fare elaborazione numerica. (N.B. il nuovissimo Quadra 610, almeno nella versione importata in Italia, possiede il coprocessore, ha il clock a 25MHz e dovrebbe dare gli stessi risultati del Centris 650).

La prova su un 486 con Windows è fortemente influenzata dalla attività di swapping e dal tipo di scheda video, inibire lo scrolling induce un netto miglioramento. I dati di tempo non erano stabili, quelle indicate sono le medie su vari tentativi.

La prova del NeXT (di cara memoria, pace all'anima sua), fatta di sfuggita a Rotterdam sulle macchine della conferenza del 1992, mostra che a parità di processore (68040 a 33 MHz) *Mathematica* gira(va) molto meglio sul NeXT, probabilmente grazie alla buona strutturazione del sistema Mach e del fatto che alla Wolfram usa(va)no soprattutto i NeXT come macchine per lo sviluppo. Sarebbe interessante provare *Mathematica* su NeXTStep per Intel (magari Pentium), datemi tempo e poi vedremo.

La prova del Pentium fa molto ben sperare per quello che sarà la velocità di queste macchine una volta disponibile una versione di *Mathematica* compilata *ad hoc*.

Favolosa la prestazione delle workstation HP Apollo (in uso monoutente) che affollavano il laboratorio dell'ESIEE di Parigi concesso per l'occasione del workshop 1993 di *Mathematica* del quale si parla in uno specifico riquadro in queste stesse pagine.

Test specifici su una sola macchina

Per avere un'idea più precisa del costo delle varie operazioni, usando *Mathematica* si è affrontato il problema dell'inversione di matrici di vario tipo su una stessa macchina: un Centris 650 (68040 a 25 MHz, 24 Megabyte di RAM, sistema 7.1) con *Mathematica* 2.2 Enhanced con 3.5 Mb per il Front-

End e 16 Mb per il Kernel. I problemi sono gli stessi del benchmark precedente, ma con dimensione variabile.

n	INVS	INV	INVMP	INVR
3	0.63			
5	24.30	0.35	0.66	0.11
7	484.95	0.36	3.43	0.26
9		0.45	9.05	0.56
11		0.80	4.73	1.03
13		0.91	20.88	1.93
15		1.45	22.21	3.46
17		1.58	27.88	5.58
19		1.98	43.48	8.75
21		2.35	62.81	14.60
23		3.46		23.33
25		9.98		36.73


Confronto con programmi Pascal

Per confrontare l'efficienza di *Mathematica* con quella di normali Pascal si è scelto il problema della costruzione e della inversione di una matrice in precisione di macchina. Il compilatore Pascal era il Think 4.0.2 e la versione di *Mathematica* la 2.2, entrambi su Centris 650. La matrice era la solita matrice della trasformata di seni: $\{ \sin(\pi i j / (n+1)) \}_{i,j, i=1,2,\dots,n, j=1,2,\dots,n}$

Si è preso in considerazione, per dimensioni da 10 a 100, passo 10, il tempo di costruzione (indicato con CP in Pascal e CM in *Mathematica*) e il tempo di inversione della matrice (IP e IM, rispettivamente). La precisione ottenuta (controllata attraverso la matrice dei residui I-AA⁻¹) era di circa 16 cifre decimali in Pascal (tipo double) e di 19 cifre in *Mathematica*. La routine di inversione (scritta da me) è un metodo di Gauss senza particolari tentativi di ottimizzazione (vedi tabella B).

Si nota che la costruzione è molto più costosa in *Mathematica* che in Pascal. Ciò si spiega perché il Pascal è un linguaggio compilato e *Mathematica* è interpretato. Sorprendentemente però l'inversione vera e propria, che in Pascal rappresenta la parte più costosa dell'elaborazione, è molto più efficiente in *Mathematica*. Non c'è alcuna contraddizione in ciò, il modulo di inversione usato in *Mathematica* è stato scritto in C, da professionisti, ottimizzato con la massima cura e compilato una volta per tutte. È naturale che sia più efficiente di una routine scritta lì per lì senza particolari pretese. I fanatici della ottimizzazione tengano presente che con Math-Link (distribuito con *Mathematica* a partire dalla versione 2.1) è possibile scrivere in C e far chiamare da *Mathematica* pezzi di codice particolarmente critici. Il mio obiettivo, quando scelgo gli strumenti per i miei esperimenti, è quello di diminuire il tempo totale che dedico alla risoluzione dei miei problemi, non di sfruttare al meglio la macchine e quindi, per ora, non mi interesso a questa possibilità.

Conclusioni

Tirare fuori una morale da questa sfilza di numeri non è facile. Spero di avervi dato un'idea di quale sarebbe la potenza di *Mathematica* su una macchina simile alla vostra. E spero anche che sia più chiaro quanto ci possa guadagnare in velocità e perdere in tempo a lavorare in linguaggi a più basso livello. Se qualcuno prova il mio benchmark (senza modificarlo in alcun modo) su macchine diverse da quelle che ho trovato io, e poi mi spedisce per E-mail i risultati sarò lieto di pubblicare en passant aggiornamenti alle tabelle viste sopra. 

Francesco Romani è raggiungibile tramite Internet all'indirizzo romani@di.unipi.it

n	CP	IP	CM	IM
10	0.00	0.03	0.23	0.20
20	0.02	0.22	0.50	0.13
30	0.05	0.65	1.08	0.31
40	0.07	1.53	1.81	0.55
50	0.12	2.93	2.76	0.933
60	0.17	4.98	3.96	1.38
70	0.22	7.88	5.25	1.95
80	0.32	11.68	6.85	2.65
90	0.37	16.58	8.45	3.53
100	0.47	22.60	10.35	4.48

Tabella B

PERSONAL 386-486

UNITÀ BASE

contenitore Desktop o MiniTower (200W), tastiera italiana estesa, controller IDE per 2 HD/2 FDD, 2 porte seriali, 1 porta parallela, 1 porta joystick, 4MB RAM, floppy disk drive 3"1/2 1.44MB, manuali tecnici di tutti i componenti.

386DX 40MHz Amd 128KB CACHE L. 880.000
486DLC 40MHz Cyrix 8KB CACHE con coprocessore L. 1.000.000



Genoa 486 Motherboard

3 slot Vesa Local Bus operanti fino a 50MHz con supporto Bus Mastering - 256KB CACHE - AMI BIOS - aggiornabile Pentium Overdrive

486S 33MHz Cyrix L. 1.150.000
486DX 33MHz Intel L. 1.520.000
486DX 40MHz Amd L. 1.560.000
486DX2 50MHz Intel L. 1.600.000
486DX 50MHz Intel L. 1.800.000
486DX2 66MHz Intel L. 1.930.000

Contenitore Tower + L. 100.000
 2° drive 5"1/4 1.2MB L. 110.000
 Controller Hard disk IDE cache 2MB, esp. 4/16MB VLB (50MHz) L. 520.000
 Microsoft DOS 6.0 italiano L. 95.000
 Microsoft DOS 6.0 + Windows 3.1 italiano L. 210.000
 SIMM 1MB/4MB 70ns L. 95.000 / L. 340.000

HARD DISK

170MB 13ms, 1.7MB/s Western Digital L. 390.000
 250MB 13ms, 1.7MB/s Western Digital L. 450.000
 340MB 13ms, 1.7MB/s Western Digital L. 560.000
 420MB 13ms, 1.7MB/s Western Digital L. 740.000
 560MB 10ms, 1.8MB/s Micropolis L. 1.200.000

FLOPTICAL 3"1/2 21MB, controller + 2 dischi + sw L. 750.000
 CONTENITORE HARD DISK AT BUS ESTERNO collegato tramite porta parallela, autoalimentato, borsa per il trasporto L. 198.000



STREAMER COLORADO

DJ-20 120/250MB

collegamento come 3° floppy disk, velocità trasferimento dati 500KB/s-1MB/s, dimensione 5"1/4, velocità backup 9.5MB/min

L. 400.000



MITSUMI CD-ROM interno MPC e Photo-CD multisezione compatibile, 350ms, 150 KB/s, 32KB buffer

L. 360.000

IN OMAGGIO un programma Windows per leggere e visualizzare i PHOTO-CD

SCHEDE VGA

Realtek 256KB 800x600 16 col. L. 50.000
 Realtek 1MB 1024x768 256 col. L. 100.000
 Cirrus Logic 5426 1MB True Color, Windows Accelerator L. 200.000
Genoa CL5426 1MB esp. 2MB True Color, VLB (50MHz) L. 270.000
Genoa M5 Veloce CL5426 + Controller HD 32 bit, VLB (50MHz) L. 370.000
Genoa ET4000 W32i 1 MB, True Color, VLB (50MHz) L. 470.000
DIAMOND VIPER WK9000 2MB VRAM, True Color, VLB (50MHz) L. 950.000

ATI Graphics Ultra Plus 2MB RAM True Color, 1280x1024i L. 580.000

ATI Graphics Ultra Pro 2MB VRAM True Color, 1280x1024ni L. 780.000

ATI Graphics Ultra Pro 2MB VRAM Vesa Local Bus, True Color, 1280x1024ni L. 800.000



MGA ULTIMA 2MB VRAM, 64 bit processor, True Color, 1280x1024ni, virtual desktop 1600x1200

ISA BUS 42M WinMark L. 1.650.000

VL BUS 60M WinMark L. 1.800.000



matrox

MONITOR

14" SRC SVGA 31.5-38KHz dot 0.28 mm L. 460.000
 14" SRC 31.5-53KHz dot 0.28 mm L. 530.000
 14" SAMPO 31.5-56KHz dot 0.28 mm L. 680.000
 15" KFC flat 30-60KHz dot 0.28 mm, controllo digitale L. 860.000
 17" MAG trinitron 30-64KHz dot 0.26 mm, controllo digitale L. 1.750.000
 20" TATUNG 31.5-64KHz dot 0.31 mm, controllo digitale L. 2.350.000
 NEC 3FGe L. 1.040.000
 NEC 6FG L. 3.990.000

486 NOTEBOOK



S-4660 Intel 486SX 25MHz
Se-4660 Cyrix 486S 33MHz
D-4660 Intel 486DX 33MHz
De-4660 Intel 486DX2 50MHz
D2-4660 Intel 486DX2 66MHz

CPU SOSTITUIBILE.

4MB RAM BASE ESPANDIBILI A 8MB.

HARD DISK REMOVIBILE 120MB e 250MB.

DISPLAY REMOVIBILE E SOSTITUIBILE:

- LCD 9.4" VGA 640x480 64 livelli di grigio
 - COLORE TFT 8.4" VGA 640x480 512 colori.

CONNETTORE BOX di espansione per schede ISA.

- Peso: 2,5 Kg con accumulatori e hard disk.
- Dimensioni: 280x224x43 mm.
- Floppy disk drive interno 3"1/2 1.44MB removibile.
- Risoluzione grafica su monitor esterno 640x480 e 800x600 256 colori, 1024x768 16 colori.
- Connettore BUS AT per BOX (305x280x50 mm) di espansione per 2 schede ISA a 16 bit + 1 unità HD 3"1/2 con alimentatore interno 100/240V.
- Tastiera micro-switch 86 tasti. Modello italiano o USA a scelta.
- Ricarica accumulatori rapida in 2 ore con notebook in funzione.
- 2 porte seriali 9 pin.
- 1 porta parallela 25 pin / connettore FDD 5"1/4 1.2MB esterno.
- Connettore standard monitor esterno 15 pin.
- Connettore tastiera esterna standard PS/2 e AT.
- Adattatore di rete 100/240V autoswitch, borsa per il trasporto.
- Dischi driver e utility scheda grafica SVGA.



LCD staccabile



Hard disk removibile



Modulo RAM



Coll. box di espansione

Unità 4660: 4MB RAM, HD 120MB, LCD b/w

S-4660 486SX 25MHz Intel L. 2.600.000
Se-4660 486S 33MHz Cyrix L. 2.650.000
D-4660 486DX 33MHz Intel L. 2.950.000
De-4660 486DX2 50MHz Intel L. 3.050.000
D2-4660 486DX2 66MHz Intel L. 3.400.000

Opzione Display colore TFT + L. 2.800.000
 Opzione Hard Disk 250MB + L. 350.000
 Hard Disk 120MB aggiuntivo L. 550.000
 Hard Disk 250MB aggiuntivo L. 900.000
 Modulo espansione 4MB RAM L. 450.000
 Docking station 2 slot ISA 16 bit L. 400.000
 Adattatore alimentaz. per auto L. 75.000
 Pacco accumulatori aggiuntivo L. 75.000
 Trackball Genius HiPoint L. 90.000
 PCMCIA unit L. 190.000

MULTIMEDIA

SOUND BLASTER Deluxe L. 150.000
 SOUND BLASTER PRO 2 Deluxe L. 220.000
 SOUND BLASTER 16 L. 310.000
 SOUND BLASTER ASP16 L. 389.000
 WAVE BLASTER per ASP16 L. 340.000

MIDI KIT per Sound Blaster L. 85.000
 VIDEO BLASTER + VIDEO per Windows L. 590.000
 VIDEO BLASTER SPIGOT + VIDEO per Windows L. 499.000
 ENCODER VGA to PAL VHS/S-VHS L. 298.000
 TV CODER PAL L. 270.000
 DISCOVERY CD-ROM + SB PRO + 5 CD L. 690.000
 PERFORMANCE CD-ROM + SB 16 + 5 CD L. 830.000
 PREMIUM CD-ROM + SB 16 + 8 CD + Voice Assist L. 960.000
 Interfaccia MIDI per portatili L. telefonare

STAMPANTI E ACCESSORI

Panasonic KX1123 24 aghi, 80 col. L. 430.000
 Panasonic KX2123 24 aghi, 80 col. L. 520.000
 Canon BJ-10sx, bubble-jet, 80 col. L. 530.000
 Canon BJ-300/330 L. 1.090.000 / 1.250.000
 HP 550C L. 1.000.000
 Genius Mouse One 2/3 tasti L. 35.000

Scanner Genius 400dpi, 256 grigi L. 300.000
 ScanMan Color Logitech, True Color L. 690.000
 Scanner A4 flat-bed 600 dpi, True Color L. 1.890.000
 MODEM/FAX CARD 2400/9600 L. 220.000
 UfoMate pocket modem/fax 2400/9600, videotel, MNP 5, windows software L. 230.000

CANON

- bubble-jet
 - 360x360 dpi
 - 240 CPS
 - introduttore fogli



BJ-200 80 colonne L. 680.000

BJ-230 136 colonne L. 940.000

PREZZI IVA ESCLUSA • COLLAUDIAMO IL MATERIALE PRIMA DELLA CONSEGNA • GARANZIA 1 ANNO SU TUTTI I COMPONENTI COMPRESIVA DI PARTI DI RICAMBIO E MANODOPERA • I COMPUTER IN CASO DI GUASTO VENGONO RITIRATI E RICONSEGNAITI TRAMITE CORRIERE A NOSTRO CARICO • SPEDIZIONI POSTALI O TRAMITE CORRIERE



Tel. (011) 3199922
 Fax (011) 3198980
 Via Piazzesi, 54/L
 10129 TORINO

Orario: Lunedì - Sabato
 9,30/13 - 15,30/19,30