A comparison of *Maple V.3* and *Mathematica 2.2.2*

Paul Zimmermann
Paul.Zimmermann@loria.fr

September 29, 1994

The following tests were done on a IBM RISC System/6000 with 64 Mb of main memory and 264 Mb of swap area. To get an idea of the speed of such a machine, the following C program takes about 5 seconds to execute (the program was compiled using `cc -O` and the time was mesured with `/bin/time`):

```
main() { int i,s=0; for (i=0;i<100000000; i++) s+=i*i; }
```

The tests are grouped together into several tables of five columns each. The first column contains the tests names, the second the Maple code, the third the Mathematica code, the fourth the time used by Maple to execute the test, and the fifth the time used by Mathematica. The times indicated are those (in seconds) given by the timing functions of both systems, `Timing[...]` for Mathematica and `st:=time(): ... time()-st` for Maple.

1 Initialization

The table below indicates the initialization time in seconds (the real time you wait to get the prompt after you have typed the command; it depends on the speed of the disk), the initial program size (SZ) and the size of the part resident in main memory (RSS) given by the Unix command `ps -xu`.

	Maple V.3	Mma 2.2
load time	0.3 s	5 s
SZ (Kb)	392	892
RSS (Kb)	828	2788

2 Rationals

	Maple code	Mathematica code	Maple	Mma
A2	`2^5000/3^5000*15^5000/10^5000`	`2^5000/3^5000*15^5000/10^5000`	3.1	0.5
D1	`s:=1:for i to 12 do s:=(s+2/s)/2 od`	`s=Nest[(#+2/#)/2&,1,12]`	0.6	0.6
D2	`s:=0: for i to 700 do s:=s+1/i od:`	`s=0;For[i=1,i<=700,i++,s=s+1/i]`	0.4	0.9

3 Reals

	Maple code	Mathematica code	Maple	Mma
	`s:=0: M:=4000`	`s=0; M=4000`		
C1	`for i to M do s:=evalhf(s+sqrt(i)) od`	`Do[s=s+N[Sqrt[i],16],{i,M}]`	0.7	12.1
C2	`for i to M do s:=evalhf(s+sin(i)) od`	`Do[s=s+N[Sin[i],16],{i,M}]`	0.7	3.2
C3	`for i to M do s:=evalhf(s+ln(i)) od`	`Do[s=s+N[Log[i],16],{i,M}]`	0.8	3.0
	`Digits:=33: s:=0: M:=250`	`s=0; M=250`		
C4	`for i to M do s:=s+evalf(sqrt(i)) od`	`Do[s=s+N[Sqrt[i],24],{i,M}]`	0.5	1.1
C5	`for i to M do s:=s+evalf(sin(i)) od`	`Do[s=s+N[Sin[i],24],{i,M}]`	2.0	1.8
C6	`for i to M do s:=s+evalf(ln(i)) od`	`Do[s=s+N[Log[i],24],{i,M}]`	1.8	1.6
A3	`evalf(exp(Pi*sqrt(163)),600)`	`N[Exp[Pi*Sqrt[163]],600]`	0.4	1.1
	`a:=evalf(Pi/E,1000):`	`a=N[Pi/E,1000];`		
L1	`evalf(sin(a),1000)`	`N[Sin[a],1000]`	1.5	1.3
L2	`evalf(ln(a),1000)`	`N[Log[a],1000]`	0.7	1.9

In Maple, the tests C1 to C3 were done with the `evalhf` function that uses machine floating-point numbers, that is objects of type *double* in the C language, whereas the `evalf` function uses Maple floating-point arithmetic. The Maple error on the sum $\sum_{i=1}^{4000} \sqrt{i} + \sin(i) + \log(i) \sim 197868.7302075302$ computed by the tests C1 to C3 is 2 units on the last digit. In Mathematica, numerical evaluations with precision less or equal to 16 also seem to use floating-point numbers of the machine; the error on the same sum is 4 on the last digit.

The tests C4 to C6 were intended to compute an approximation of the sum $\sum_{i=1}^{250} \sqrt{i} + \sin(i) + \log(i) \sim 3777.186475204937271158413317407$3 accurate to 32 decimal places (that is an error of at most 1 unit on the last digit). In Maple, we had to set the variable `Digits` to 33 to get this accuracy. In Mathematica, a precision of 24 was enough (but 23 was not).

4 Solving equations

4.1 Linear algebra

	Maple code	Mathematica code	Maple	Mma
	`M:=16: A:=array(1..M,1..M):`	`M=16`		
	`for i to M do`	`f[n_]:=Map[n^#&,Range[M]]`		
	`  for j to M do A[i,j]:=i^j od`	`A = Map[f,Range[M]]`		
	`od`			
N1	`linalg[inverse](A);`	`Inverse[A]`	4.9	23.1
	`M:=5; L := {seq(x[i],i=1..M)}`	`M=5; L = x[#]& /@ Range[M]`		
	`S:={seq(sum(x[i]/(a+i+j),`	`S=Sum[x[i]/(a+i+#),{i,1,M}]==#^2&`		
	`    i=1..M)=j^2,j=1..M)}:`	`  /@ Range[M]`		
N2	`solve(S,L)`	`Solve[S,L]`	1.3	6.1

4.2 Numeric roots of complex polynomials

$$p_1 = 2 + 3x + 5x^2 + 7x^3 + 11x^4 + 13x^5 + 17x^6 + 19x^7 + 23x^8 + 29x^9$$
$$p_2 = p_1 + 31x^{10} + 37x^{11} + 41x^{12} + 43x^{13} + 47x^{14} + 53x^{15} + 59x^{16} + 61x^{17} + 67x^{18} + 71x^{19}$$
$$p_3 = p_2 + 73x^{20} + 79x^{21} + 83x^{22} + 89x^{23} + 97x^{24} + 101x^{25} + 103x^{26} + 107x^{27} + 109x^{28} + 113x^{29}$$

	Maple code	Mathematica code	Maple	Mma
	`Digits:=28;`			
N3	`fsolve(p1+I,x,complex);`	`NRoots[p1+I==0,x,20]`	1.5	2.7
N4	`fsolve(p2+I,x,complex);`	`NRoots[p2+I==0,x,20]`	2.8	11.7
N5	`fsolve(p3+I,x,complex);`	`NRoots[p3+I==0,x,20]`	5.7	27.3

For the test N3, the maximal modulus when one replaces the given roots in $p_1 + I$ is $2 \cdot 10^{-27}$ for Maple and $5 \cdot 10^{-31}$ in Mathematica. For the test N4, the maximal errors are $2 \cdot 10^{-26}$ for Maple and $4 \cdot 10^{-29}$ in Mathematica. For N5, the errors are $8 \cdot 10^{-26}$ for Maple and $8 \cdot 10^{-28}$ in Mathematica. As we already noticed before, we had to ask a higher precision of Maple, in order to get the same accuracy at the end.

5 Control

	Maple code	Mathematica code	Maple	Mma
B1	`for i to 5000 do`	`For[i=1,i<=5000,i++,`		
B1	`  if i mod 2=0 then s:=s+i+1`	`  If[Mod[i,2]==0,s=s+i+1,s=s-i]`	0.8	4.2
B1	`  else s:=s-i fi od`	`]`		
	`fb:=proc(n) if n<=1 then 1`	`fb[n_] := If[n<=1, 1,`		
	`  else 1+fb(n-1)+fb(n-2) fi end`	`  1+fb[n-1]+fb[n-2]]`		
B2	`fb(18)`	`fb[18]`	1.1	5.4
	`f:=proc(n) option remember; n end`	`f[n_]:=f[n]=n`		
B3	`s:=0: for i to 3000 do s:=s+f(i) od`	`s=0; Do[s=s+f[j],{j,3000}]`	0.8	5.1
	`h := proc(n) option remember;`	`$RecursionLimit=Infinity`		
	`  h(n-h(n-1)) + h(n-h(n-2)) end`	`h[n_]:=h[n]=h[n-h[n-1]]+`		
	`h(1):=1: h(2):=1`	`   h[n-h[n-2]]; h[1]=h[2]=1`		
B4	`h(1000)`	`h[1000]`	0.4	2.8

Note: test B4 induces a very high depth of recursive calls.

6 Symbolic computations

	Maple code	Mathematica code	Maple	Mma
A1	`series(1/cos(z),z=0,51)`	`Series[1/Cos[z],{z,0,50}]`	0.5	5.2
	`f:=exp(cos(sqrt(1-x^4)))`	`f=Exp[Cos[Sqrt[1-x^4]]]`		
D3	`series(f,x=1/sqrt(2),9): evalf(")`	`Series[f,{x,1/Sqrt[2],8}]; N[%]`	0.5	7.4
A4	`to 120 do d:=1/(1+x^4);`	`Do[d=1/(1+x^4);`		
A4	`  for i to 10 do d:=diff(d,x) od`	`  Do[d=D[d,x],{10}],`	0.5	14.8
A4	`od`	`{120}]`		
A5	`for i to 10 do d:=int(d,x) od;`	`Do[d=Integrate[d,x],{10}]`	1.8	2.0
	`p:=expand(product(product(`	`p=Expand[Product[Product[`		
	`  x[i]-x[j],i=j+1..5),j=1..5)):`	`  x[i]-x[j],{i,j+1,5}],{j,1,5}]]`		
D4	`factor(p)`	`Factor[p]`	0.5	0.6
D5	`expand(radsimp(1/(sqrt(2)+sqrt(3)`	`Apart[1/(Sqrt[2]+Sqrt[3]+Sqrt[5]`		
D5	`  +sqrt(5)+sqrt(7)+sqrt(11)`	` +Sqrt[7]+Sqrt[11]+Sqrt[13])]`		
D5	`  +sqrt(13)),'ratdenom'));`		0.6	9.0

Note: in Maple, the third argument of `series` is the order of the big-O term. Thus, to get an expansion of n terms, you have to give $n + 1$ as third argument.

7 Arithmetic and Control

	Maple code	Mathematica code	Maple	Mma
E1	`ee:=sum(1/i!,i=0..99)`	`ee=Sum[1/i!,{i,0,99}]`	0.4	1.2
	`CF:=proc(x,n)`	`CF[x_, n_Integer?Positive]:=`		
	`local xi,xp,i;`	`  Block[{xi,xp=x,r={}},`		
	`  xp := x;`	`    Do[`		
	`  xi := array(1..n);`	`      xi=Floor[xp];`		
	`  for i to n do`	`      AppendTo[r,xi];`		
	`    xi[i]:=trunc(xp);`	`      xp=1/(xp-xi),`		
	`    xp:=1/(xp-")`	`    {n}];`		
	`  od;`	`    Return[r]`		
	`  op(xi)`	`  ]`		
	`end:`			

	Maple code	Mathematica code	Maple	Mma
E2	`CF(ee^5,500)`	`CF[ee^5,500]`	0.5	1.6

8 Taylor series expansions

	Maple code	Mathematica code	Maple	Mma
	`u[1]:= x -> log(1/(1-x))`	`u[1,x_] := Log[1/(1-x)]`		
	`u[2]:= x -> x/(1+x)`	`u[2,x_] := x/(1+x)`		
	`u[3]:= x -> exp(x)-1`	`u[3,x_] := Exp[x]-1`		
	`u[4]:= x -> sqrt(1-x)-1`	`u[4,x_] := Sqrt[1-x]-1`		
	`u[5]:= x -> (1+x)^(1/3)-1`	`u[5,x_] := (1+x)^(1/3)-1`		
	`u[6]:= x -> (1-x)*(1-x^2)*(1-x^3)*`	`u[6,x_]:=(1-x)(1-x^2)(1-x^3)`		
	`          (1-x^4)*(1-x^5)-1`	`          (1-x^4)(1-x^5)-1`		
	`u[7]:= x -> x+x^2+x^3+x^4+x^5`	`u[7,x_]:=x+x^2+x^3+x^4+x^5`		
	`u[0]:= x -> sin(x)`	`u[0,x_]:=Sin[x]`		
	`test:=proc(n)`	`Test[n_]:=Block[{},`		
	` t:=x;`	` t:=x;`		
	` for i to 8*n do`	` Do[`		
	`  t:=series(u[i mod 8](t),x=0,13)`	`  t=Series[u[Mod[i,8],t],`		
	` od;`	` {x,0,12}],{i,8n}];`		
	` t`	`Return[t]`		
	`end:`	`]`		
F1	`test(1)`	`Test[1]`	0.7	5.7
F2	`test(2)`	`Test[2]`	0.9	7.7

9 Differentiation

	Maple code	Mathematica code	Maple	Mma
	`test:=proc(nn)`	`Test[nn_]:=Block[{},`		
	`  s:=0;`	`  s=0;`		
	`  for n to nn do`	`  Do[`		
	`    d:=diff(sin(x)^n,x$n);`	`    d=D[Sin[x]^n,{x,n}];`		
	`    s:=s+subs(x=0,d)*z^n;`	`    s=s+(d /. x->0)*z^n,`		
	`  od;`	`  {n,1,nn}];`		
	`  s`	`  Return[s]`		
	`end;`	`]`		
G1	`test(30);`	`Test[30]`	0.5	18.7

10 Factorization

10.1 Modular factorization

$$a := 78 + 17x + 72x^2 - 99x^3 - 85x^4 - 86x^5 + 30x^6 + 80x^7 + 72x^8 + 66x^9 - 29x^{10} - 91x^{11} - 53x^{12} - 19x^{13} - 47x^{14} + 68x^{15} - 72x^{16} - 87x^{17} + 79x^{18} + 43x^{19} - 66x^{20} - 53x^{21} - 61x^{22} - 23x^{23} - 37x^{24} + 31x^{25} - 34x^{26} - 42x^{27} + 88x^{28} - 76x^{29} - 65x^{30} + 25x^{31} + 28x^{32} - 61x^{33} - 60x^{34} + 9x^{35} + 29x^{36} - 66x^{37} - 32x^{38} + 78x^{39} + 39x^{40} + 94x^{41} + 68x^{42} - 17x^{43} - 98x^{44} - 36x^{45} + 40x^{46} + 22x^{47} + 5x^{48} - 88x^{49} - 43x^{50};$$

	Maple code	Mathematica code	Maple	Mma
		`T[p_] := Factor[a,Modulus->p]`		
I1	`for p in {2,3,5,7}`	`Map[T,{2,3,5,7}]`	0.4	0.9
I1	`do Factor(a) mod p od:`			
I2	`for p in {101,103,107,109}`	`Map[T,{101,103,107,109}]`	0.9	1.6
I2	`do Factor(a) mod p od:`			
I3	`for p in {10007,10009,10037,10039}`	`Map[T,{10007,10009,10037,10039}]`	4.7	1.8
I3	`do Factor(a) mod p od:`			

10.2 Factorization of composite polynomials

We consider here the polynomials

$$\begin{aligned} p_2 &= \left(4 - 4x + 3x^3\right)\left(-6 + 7x - 7x^2 + 5x^3\right)\left(-7 + 4x - 7x^2 - 5x^3 + 8x^4\right)\left(-3 - 6x - 7x^2 + 6x^3 + 3x^4\right) \\ p_6 &= \left(x^{25} + x^{17} + 11x^5 + 12\right)\left(x^{26} + 47x^{18} + 19x^6 + 81\right) \\ p_7 &= \left(x^{33} - 1\right)\left(x^{31} - 1\right) \end{aligned}$$

and b_n, the expanded product of the Bernoulli polynomials of order n and $n+1$.

	Maple code	Mathematica code	Maple	Mma
H1	`factor(p2)`	`Factor[p2]`	0.5	0.2
H2	`factor(p6)`	`Factor[p6]`	1.0	0.8
H3	`factor(p7)`	`Factor[p7]`	1.0	1.0
J1	`factor(b10)`	`Factor[b10]`	0.4	0.3
J2	`factor(b20)`	`Factor[b20]`	1.6	2.6
J3	`factor(b30)`	`Factor[b30]`	5.4	10.0
J4	`factor(b40)`	`Factor[b40]`	9.2	36.4

10.3 Factorization of almost irreducible polynomials

	Maple code	Mathematica code	Maple	Mma
	`b25:=bernoulli(25,x)`	`b25 = BernoulliB[25,x]`		
	`b26:=bernoulli(26,x)`	`b26 = BernoulliB[26,x]`		
K1	`factor(b25): factor(b26)`	`Factor[b25]; Factor[b26]`	0.4	0.6
K2	`factor(b45): factor(b46)`	`Factor[b45]; Factor[b46]`	0.5	2.9
K3	`factor(b65): factor(b66)`	`Factor[b65]; Factor[b66]`	1.4	8.3
K4	`factor(b85): factor(b86)`	`Factor[b85]; Factor[b86]`	3.5	70.3

11 Plots

	Maple code	Mathematica code	Maple	Mma
M1	`plot(log(1+sin(tan(x))),`	`Plot[Log[1+Sin[Tan[x]]],`		
M1	`x=0..Pi/2,numpoints=100)`	`{x,0,Pi/2},PlotPoints->100]`	0.4	0.3
M2	`plot([sin(3*t),sin(5*t),`	`ParametricPlot[{Sin[3t],Sin[5t]},`		
M2	`t=0..2*Pi],numpoints=100)`	`{t,0,2Pi},PlotPoints->100]`	1.4	0.5
M3	`plot3d(sin(x)*sin(y),x=-10..10,`	`Plot3D[Sin[x] Sin[y],{x,-10,10},`		
M3	`y=-10..10,grid=[40,40],`	`{y,-10,10},PlotPoints->40]`	0.5	2.2
M3	`orientation=[-62,36])`			

Note: the example M3 was taken from page 122 of the Mathematica reference manual, first edition.

12 Conclusions

The table below compares the efficiency of both systems for each type of operation. A single + in a column means that the corresponding system is between 1 and 2 times faster than the other one, a ++ for between 2 and 4 times faster, and a +++ for more than 4 times faster (we considered the geometric mean of the time ratios in the case of several tests).

	Maple V.2	Mma 2.2
Rationals		+
Reals 16 digits	+++	
Reals 32 digits	+	
Reals 600-1000 digits	+	
Rational linear algebra	+++	
Symbolic linear algebra	+++	
Numeric complex roots	++	
Control	+++	
Symbolic	+++	
Arithmetic/control	++	
Taylor series	+++	
Differentiation	+++	
Factor (modular, $p < 10^4$)	+	
Factor (modular, $p > 10^4$)		++
Factor (composite)	+	
Factor (irreducible)	+++	
Plot 2D		+
Plot 3D	+++	

Acknowledgment: I want to thank Philippe Flajolet who made the first version of this comparison, and Greg Fee who made several comments on a previous version. Test B4 (Hoftadter's function) was suggested by Daniel Pfenniger (University of Geneva). Test D5 was suggested by John Abott. Many thanks to the GDR-Medicis, specially to the group at École Polytechnique (Palaiseau, France) who gave me access to the machine where these tests were done.