

muMATH

David R. Stoutemyer, 28.1.2012:

I am delighted that someone is keeping alive the memory of muMath!
best regards, david

Computer-Algebra-System für CP/M

muMATH (manchmal auch myMATH geschrieben) war seinerzeit ein beeindruckendes Stück Software: Ein Programm, das mit Zahlen beliebiger Länge rechnen konnte, Gleichungen mit Variablen algebraisch löste u.v.m. Und das alles mit den beschränkten Speicher- und Rechenkapazitäten von CP/M!

muMATH ist fähig, mathematische Formelausdrücke umzuformen. Dabei ist es im Unterschied zu „klassischen“ Programmiersprachen, wie ALGOL, FORTRAN, PL/I, PASCAL, BASIC usw. nicht notwendig, dass die verwendeten Variablen zur Laufzeit einen Wert besitzen. Variablen werden, wie in der Mathematik üblich, als formale Rechengrößen verwendet, sie können als Bezeichner für einen Ausdruck fungieren, sie können aber auch für sich selbst stehen.

muMATH wurde in den 70er Jahren von von Albert D. Rich und David Ross Stoutemyer entwickelt. Beide gründeten 1979 das Unternehmen The Soft Warehouse.

Zur Computeralgebrageschichte und zu Implementationsdetails hat Stoutemyer 2008 einen Vortrag gehalten:

http://www.orcca.on.ca/conferences/cca2008/presentations/Stoutemyer_WaysToImplementComputerAlgebraCompactly.pdf

muMATH ist in einem LISP-Derivat namens muSIMP geschrieben. muMATH war das erste Computeralgebrasystem (CAS), das auf „normalen“ kleinen Computern lief! Alle bis dahin verfügbaren Computeralgebrasysteme waren nur für Großrechner geschrieben. Der Namensanfang mu (my, μ) steht hier also synonym für mikro-Programmgröße und Mikro-Prozessoren.

muMATH-79 erschien 1979 und lief auf 8080- und Z80-Computern mit weniger als 48Kbyte RAM unter CP/M, und auf dem Radio Shack TRS-80 unter TRS-DOS. muMATH-80 erschien 1980 und lief zusätzlich auf 6502 based Apple II Computern. muMATH-83 erschien 1983 und lief auf 8088 based IBM PC und XT Computern mit weniger als 300Kbytes RAM.



muSIMP/muMATH wurde von Microsoft zum Preis von 250 US-\$ vertrieben (im Programm steht muSIMP-83 2.02 COPYRIGHT (C) 1980 MICROSOFT LICENSED FROM THE SOFT WAREHOUSE). Nebenstehendes Bild zeigt die zugehörige Seite eines Microsoft-Katalogs, ca. 1980.

<http://channel9.msdn.com/shows/History/The-History-of-Microsoft-1982/> schreibt außerdem: *September 1, 1982: Microsoft announces the availability of its symbolic mathematic package muMATH/muSIMP for the IBM Personal Computer. This package is also offered for the Apple II, TRS-80, and CP/M-80 computer systems.*

DERIVE, der Nachfolger von muMATH, erschien 1988 und war unter DOS und später Windows lange Zeit gerade in der schulischen und studentischen Ausbildung verbreitet.

Downloads

- [mymath.pdf](#) deutsche Anleitung der TUK (Dank an J. Gollasch!)
- [mumath.zip](#) verschiedene Versionen: originale Versionen muMATH-83 V2.02, V2.03, V2.10, V2.12, V2.14, deutsche Version der TUK (entspricht V2.12)
- http://www.retroarchive.org/docs/mumath_musimp.pdf engl. Anleitung (Achtung 15 MByte!)
- [cpmug083_mummath.zip](#) Erweiterungen zu muMATH v2.12 von Gerald A. Edgar, veröffentlicht auf der CPMUG volume 83.

Links

- http://metaresearch.com/a/in_praise_of_musimp.html
- [Capabilities of the MUMATH-78 computer algebra system for the INTEL-8080 microprocessor](#)
- [PICOMATH-80™: an even smaller computer algebra package](#)

Schnellstart

Zum Kennenlernen von Musimp (Version 2.14):

1. Starte Musimp am CP/M prompt: **MUSIMP ALL**
2. Setze „PAUSE“ für die nachfolgende Demo durch folgende Eingabe am „?“ Prompt:
PAUSE:100;
3. Starte die Demo durch folgende Eingabe am „?“ Prompt: **RDS(DEMO,ALL);**
4. Beende Musimp

```
%***  INTEGER ARITHMETIC EXAMPLES  ***%
```

```
% INTEGER ADDITION & SUBTRACTION %
```

```
32 + 15 - 24;
```

```
@: 23
```

```
? % MULTIPLICATION & UNARY MINUS %
```

```
5 * -12;
```

```
@: -60
```

```
? % USE OF PARENTHESIS %
```

```
436 * (123 - 57);
```

```
@: 28776
```

? % RAISING TO A POWER %

100^3;
@: 1000000

? % ASSIGNMENTS TO A VARIABLE %

F00: (3*8 - 16)^2;
@: 64

? % USE OF ASSIGNED VARIABLE %

3*F00^5;
@: 3221225472

? % SAVE INTERMEDIATE RESULTS %

SEC#PER#YR: 365*24*3600;
@: 31536000

? IN#PER#MI: 5280*12;
@: 63360

? % USE OF INTERMEDIATE RESULTS %
% TO FIND INCHES TO ALHPA CENTAURI %

4 * 186000 * SEC#PER#YR * IN#PER#MI;
@: 1486601994240000000

? % EXACT, INFINITE PRECISION %

99^99;
@:
3697296376497267726571879056288054405956687642817411
0243025997242355257045527752342141065001012823272794
0978889548326540119429996769494359451621570193644014
418071060667659301384999779999159200499899

?
%*** RATIONAL ARITHMETIC EXAMPLES ***%

% REDUCE FRACTIONS TO LOWEST TERMS %

56/77;

@: 8 / 11

? % FIND COMMON DENOMINATOR %

$5/6 - 3/4;$

@: 1 / 12

? % RATIONAL SIMPLIFICATION %

$3 * (1/2 + 1/6);$

@: 2

? % FLOATING POINT NOTATION %

POINT: 10\$

? 1/3;

@: 0.3333333333

?

%*** VARIABLE RADIX BASE ***%

% SET FOR HEXADECIMAL ARITHMETIC %

RADIX (16);

@: 0A

? % USE AS A HEX CALCULATOR %

$7C80 - 2*12EF + 0A3C;$

@: 60DE

? % ASSIGNMENT TO A VARIABLE %

EG: 10000;

@: 10000

? % RETURN TO BASE TEN ARITHMETIC %

RADIX (0A);

@: 16

? % FIND EG IN BASE TEN %

EG;
@: 65536

? % BASE TWO ARITHMETIC %

RADIX (2);
@: 1010

? % BINARY ARITHMETIC CALCULATOR %

101101110 * EG;
@: 101101110000000000000000

? % RETURN TO BASE TEN ARITHMETIC %

RADIX (1010);
@: 2

?
%*** EXPONENTIAL SIMPLIFICATIONS ***%

% FRACTIONAL POWERS %
8 ^ (2/3);
@: 4

? 12 ^ (1/2);
@: 2 * 3^0.5

? % POWERS OF THE IMAGINARY NUMBER %

#I^2;
@: -1

? #I^-7;
@: #I

? % COMPLEX EXPONENTIALS %

#E ^ (#I*#PI);
@: -1

?

***%

```
5!;  
@: 120
```

```
? 50!^2;  
@:  
9250170652825079190134707232358836823494868074219019  
8770613927101881057071736043444238321314044821530214  
400000000000000000000000000000
```

? % BINOMIAL COEFFICIENTS [12:30] %

```
N: 30;  
@: 30
```

```
? M: 12;  
@: 12
```

```
? N! / ((N-M)!*M!);
@: 86493225
```

```
?
%*** BASIC ALGEBRA EXAMPLES ***%
```

```
% AUTOMATIC ALGEBRAIC SIMPLIFICATION %
```

% COMBINES SIMILAR TERMS AND FACTORS %

```
3*X - X;  
@: 2 * X
```

```
? Y^3 * Y^(R+1);
@: Y ^ (4+R)
```

```
?
% EXPLOITS IDENTITIES AND ZEROS %
```

```
0 + X;  
@: X
```

```
? 1 * Y;  
@: Y
```

? Z * 0;

@: 0

? X^1;

@: X

? Y^0;

@: 1

? 1^X;

@: 1

? % MULTIPLE SIMPLIFICATIONS %

5*X^1*Y + Y^2*-3*X/Y + W^(Z^2 - Z*Z);

@: 1 + 2*X*Y

? % POLYNOMIAL MULTIPLICATION %

(3*X - 2*Y) * (Y^2 + 4*X);

@: -8*X*Y + 3*X*Y^2 + 12*X^2 - 2*Y^3

? % POLYNOMIAL POWERS %

(X+1)^2;

@: 1 + 2*X + X^2

? (X+5) * (X^2-2*X+3)^2;

@: 45 - 51*X + 38*X^2 - 10*X^3 + X^4 + X^5

? % CONTENT FACTORIZATION %

FCTR (6*X^3*Y + 15*X^2*Y);

@: 3 * X^2 * Y * (5+2*X)

?

*** LOGARITHMIC SIMPLIFICATIONS ***

% NATURAL LOG OF ONE %

LN (1);

@: 0

? % COMMON LOG OF 1000 %

LOG (1000, 10);

@: 3

? % EXPAND THE LOG OF A PRODUCT %

LN (X*Y);

@: LN(X) + LN(Y)

? % EXPAND THE LOG OF A POWER %

LOG (Z^3, 10);

@: 3*LN(Z) / LN(10)

? % MULTIPLE SIMPLIFICATIONS %

LN(X^2*Y) - 2*LN(X);

@: LN (Y)

? % INTER-BASE SIMPLIFICATIONS %

LOG(X,10) * LOG(10,#E);

@: LN (X)

? % LOGARITHMIC POWERS %

#E ^ LN(X+5);

@: 5 + X

?

%*** TRIGONOMETRIC SIMPLIFICATIONS ***%

% ELEMENTARY ANGLE VALUES %

COS (0);

@: 1

? SIN (#PI/2);

@: 1

? SIN (37*#PI/3);

@: 3^0.5 / 2

? % EQUIVALENT FUNCTIONS %

TAN(X) * COS(X);

@: SIN (X)

? % MULTIPLE ANGLES EXPANSION %

SIN (2*X);

@: 2 * COS(X) * SIN(X)

? COS(3*X);

@: 4*COS(X)^3 - 3*COS(X)

? % ANGLE SUMS EXPANSION %

COS (X-Y);

@: COS(X)*COS(Y) + SIN(X)*SIN(Y)

? % COMBINATION EXPANSIONS %

EG: SIN (2*X+Y);

@: 2*COS(X)^2*SIN(Y) + 2*COS(X)*COS(Y)*SIN(X) - SIN(Y)

?

%*** REPRESENT EQUATIONS ***%

EQN: 2*X+7 == A^2 - X^2/X - 3;

@: 7+2*X == -3-X+A^2

? % STEP BY STEP SOLUTION FOR X %

EQN: EQN + X - 7;

@: 3*X == -10+A^2

? EQN: EQN/3;

@: X == (-10+A^2)/3

?

%*** CALCULUS OPERATIONS ***%

% FIND DERIVATIVES %

```
DIF (3*X^2 + 5*X - 4, X);
@: 5 + 6*X
```

```
? DIF (LN(X)^2, X);
@: 2*LN(X) / X
```

```
? DIF (#E^X^2, X);
@: 2 * #E^X^2 * X
```

```
? DIF (P*SIN(X) + X^2, X);
@: 2*X + P*COS(X)
```

```
? % FIND INTEGRALS %
```

```
INT (2*X - 1/X, X);
@: X^2 - LN(X)
```

```
? INT (X * #E^X^2 * SIN(#E^X^2), X);
@: -COS(#E^X^2) / 2
```

```
? INT (LN(LN(X))/X, X);
@: LN(X)*LN(LN(X)) - LN(X)
```

```
?
%***      PROGRAMMING IN MUSIMP      ***%
```

```
% TAYLOR SERIES EXPANSION FUNCTION %
```

```
FUNCTION TAYLOR (EXPX, X, A, N,
% Locals: % J, C, ANS, NUMNUM, DENNUM),
  NUMNUM: DENNUM: 30,
  J: ANS: 0,
  C: 1,
  LOOP
    ANS: ANS + C * EVSUB (EXPX, X, A),
    WHEN J=N, ANS EXIT,
    EXPX: DIF (EXPX, X),
    J: J + 1,
    C: C * (X-A) / J,
  ENDLLOOP,
ENDFUN ;
@: *** REDEFINED: TAYLOR
TAYLOR
```

? % TAYLOR SERIES EXPANSION %

TAYLOR (#E^X, X, 0, 6);

@: $1 + X + \frac{X^2}{2} + \frac{X^3}{6} + \frac{X^4}{24} + \frac{X^5}{120} + \frac{X^6}{720}$

? TAYLOR (SIN(X), X, 0, 8);

@: $X - \frac{X^3}{6} + \frac{X^5}{120} - \frac{X^7}{5040}$

? TAYLOR (#E^SIN(X), X, 0, 4);

@: $1 + X + \frac{X^2}{2} - \frac{X^4}{8}$

?

MOVD ('CRLF, 'NEWLINE)\$

? RDS ()\$

? ^C

From:

<https://hc-ddr.hucki.net/wiki/> - Homecomputer DDR

Permanent link:

<https://hc-ddr.hucki.net/wiki/doku.php/cpm/mumath?rev=1327828959>

Last update: **2012/01/29 09:22**

