MAD 3401 — Introductory Numerical Analysis

Section 1, Fall 1994.

Instructor: Bellenot.

The good doctor's Office: 002-B Love, Office Hours: MWF 12:30–1:15 or by appointment. Email addressed bellenot@cs.fsu.edu, bellenot@math.fsu.edu, or even bellenot@fsu.edu will get to the good doctor, but the short address 'bellenot' works on math or cs machines.

Eligibility: A grade of C- or better in Calculus 2 (MAC 3312) and knowledge of a high level programming language such as C, Pascal or FORTRAN.

Text: K. Atkinson *Elementary Numerical Analysis* 2^{nd} Edition.

Coverage: Chapters 1 - 7 and 8.6.

Final: At 10-12 Monday Dec 12, 1994.

Tests: (3) Tentatively at Sept 21, Oct 26 and Nov 30. No Makeup tests.

Programming Projects: Due roughly every other Wednesday. Most of the projects will require the use of Maple.

Grades: 90% A, 80% B, 70% C, 60% D.

Relative Weights F = 2T and T = P (F is 1/3, each T is 1/6 and P is 1/6).

Homework and Attendance are required. Indeed attendance will be taken by checking off homework. It is the student's responsibility to see that homework is delivered on time. (The homework needs to be turned in even when the student is absent.) Likewise, being absence is not a valid reason for not knowing the next assignment.

Three or more late or missing homeworks is an automatic FAIL.

You need to turn in a print out of a maple file (small fontsize -- change with options menu, output fontsize selection, small) Which contains the functions taylorg, taylorh, and taylork which are to the functions g, h, and k below as the example taylorf is to the example function f. Also your print out needs to show the results of running each taylor? function with each combination of m = 2, 5, 10 and 50 and b = 0.01, 0.1, 0.5 and (except for g) 1.0 and 2.0.

Note: taylorg will do bad things if you try b >= 1.

Due Date: 28 Sept

Project outline. Given a function, an interval and an error bound find the number of terms n so that the taylor polynominal p_n will be within the error bound on the interval for the given function.

The error bound will be given as an integer m which is to mean that the error bound <= 5 x 10^-(m+1) (The absolute error is small enough so that the first m-digits after the decimal point is correct.)

The interval will be given as a real number b > 0 which is to mean that the interval is -b <= x <= b. (And hence all our taylor poly's are for a=0.)

The functions f are given below. Note that you will need a separate procedure for each one.

For the error bounds you are to use the remainder term in taylors.

The functions:
1. g(x) = (1-x)^-1
2. h(x) = exp(x)
3. k(x) = integral from 0 to x of exp (- t^2) dt

Sample data's
The error 'm's are 2, 5, 10 and 50
The intervals 'b's are 0.5 for 1, 0.5 and 1.0 for 2 and 0.1 for 3.

Example procedure for to check when f(x) = sqrt (1 + x) has a small enuff error. The n-th derivative of f(x) is has a (1 + x) to a power in the bottom. so we will assume the interval is less than one and the largest value of the n-th derivative will occur at the point x = -b.

see the maple file proj_1.m

```
taylorf := proc ( b, m )
        local allow, current, n, enuff;
        Digits := 2 * m + 1;
        allow := 5 * 10^(-m-1);
        n := 0; # constant
        enuff := evalf(current -allow );
        current := evalf( 0.5 * b / sqrt ( 1 - b ) );
        while enuff > 0 do
                n:=n+1;
                current:=evalf( current * b * ( 2n - 1 )/2 / ( 1 - b);
                enuff := evalf(current -allow );
        od;
        n
        end;
```

Hints: 1. To continue a statement (like a procedure definition) to the next line use SHIFT-return.

2. To aid with debugging the command trace(taylorf) will show you the steps in taylorf as they execute.

For the error parameter in bisect use 10^-6 (0.000001 one-millionth)

Data to test
1. f(x) = x^3 - 4.7 x^2 + 6.84 x - 3.168 a=-4, b=4, mesh = 0.25
2. f(x) = x^3 - 4.7 x^2 + 6.84 x - 3.168 a=-4, b=4, mesh = 0.05
3. f(x) = x - e^e, a=0, b=2, mesh = 0.1
4. f(x) = sin(x), a=0, b=100, mesh = 1
5. f(x) = x^2 - 1.1x + 0.3024, a=0, b=2, mesh 0.1
6. f(x) = x^2 - 1.1x + 0.3024, a=0, b=2, mesh 0.04

Proj2's goal is roughly, given a function f(x) find all of its zero's --that is all points z so that f(z) = 0.

The first step is to find a zero by the bisection method. The bisect proc has inputs f, a, b, error where f is a function with f(a)*f(b) < -1, a < b, and 0 < error. The bisect proc returns a c so that some zero z of f satisfies |c - z| < error.

The bisection algorithm (slightly different then the one in class)

```
        c <- (a + b)/2
        while b - a > 2 * error do
        {
                if ( f(c) equal 0 )
                        return c
                else if ( f(c)*f(b) < 0 )
                        a <- c
                else
                        b <- c

                c <- (a + b)/2
        }
        return c
```

The second step is a proc allzero (f, a, b, mesh) which returns a list of all zero's z, the function f has in the interval a <= z <= b. The proc allzero is allowed to miss zero's z if both f(c) and f(d) have the same sign where d-c = mesh, c = a + j * mesh < z < d = a + (j + 1) * mesh.

Maple lists are enclosed in []; The empty list is []; the way to cancatenate lists a and b is [op(a), op(b)]; the way to append (prepend) 4 to list a is [op(a), 4] ([4, op(a)]).

So allzero should do

```
        zero <- []
        while there-is-another-zero
        {
                find A, B for bisect
                z <- bisect(f, A, B, error)
                zero <- [op(zero), z]
        }
        return zero
```

Test functions:
1 f(x) = (x - r)(x - s)(x - t)
2 f(x) = x^2 + 1
3 f(x) = (x - 1.1)^3
4 f(x) = e^-x * sin x

Preview of proj3:
The above list does not include multiplicities, at least as written. This list should include multiplicities, if z is a root twice (trice, etc) then z should appear twice (trice, etc).

Also in proj3, a newton solver to replace bisect.

First Draft Oct 19
Update 31 Oct

Multiple Roots

Real Data:
A. f(x) = x^2 starting from 1
B. f(x) = x^3 starting from 1
C. f(x) = x^10 starting from 1
D. f(x) = sin^2 (x) starting from 3
E. 4.5#1 has two roots near -1, and 1
F. 4.5#2 has three roots near -1, 1 and 2

1. A function guessm with parameters f and a. guessm uses a couple of iterations of newtons method to guess the multiplicity of the root of f near a. A return value of zero says newton's is not converging.

2. A function mnewton with parameters f, a and error; which uses the value of guessm to convert to another function g with the same root as f but as a single root and use newton to find the root.

3. A function anewton which uses Aitken extrapolates (like 4.4#13)

Sample test functions x^2 at 1, x^3 at 1, 4.5#1&2 4.1#6&7. Real data later.

Output: for each function:
for each method in newton, mnewton, anewton
output the number of iterations, the final estimate (and for mnewton the multiplcity).

We want to produce a page of 6 graphs (via plot-it.tex) and the maple code to needed to produce them. Each of the graphs is going to compare a function with a polynomial interpolate obtained by picking points.

```
        f(x) = 1/(1 + x^2)
```

fig1.ps n = 4, the data points x_0, x_1, x_2, x_3 and x_4 are equally spaced on [-5,5] and the y values are obtained by y_i = f(x_i). (Note that if you define xdata as array(1..n), then the maple n is going to be one more the given n.) Obtain a polynomial interpolate for the data, say p, and plot p and f via the command

```
        plot({f(x), p(x)}, x=-5..5, y=-1..2,
        title=`Equal spaced interpolation of f(x)=1/(1+x^2) (n=4) by Your Name`);
```

When the plot comes up, save this file as fig1.ps

fig2.ps same as above but n = 8 (and off by one in maple).
fig3.ps same as above but n = 16 (and off by one in maple).

The data points for the next three points use what is known as Chebyshev spacing. x_i = 5 * cos ((2i+1)Pi/(2n + 2)) for i = 0,1,2,...n (note both i and n will be different in maple). Still the y_i = f(x_i). [It does not matter that the x_i's are not in order.] Obtain a polynomial interpolate for this data, say p and plot p and f like the above only save it to fig4.ps. Change the title by replacing 'equal' with 'Chebyshev'

fig5.ps same as fig4.ps but with n = 8 (and off by one in maple)
fig6.ps same as fig4.ps but with n = 16 (and off by one in maple)

Copy and Read plot-it.tex to obtain a file containing all 6 plots in one ps file.

The old hint file.

```
> ?vec
Try one of the following topics:


                          {vecpotent, vectdim, vector}
--------------------------------------------------------------------------------
> ?vector
--------------------------------------------------------------------------------
> n:=11;

                                    n := 11
--------------------------------------------------------------------------------
> x:=array(1..n);

                             x := array(1 .. 11, [])
--------------------------------------------------------------------------------
> for i from 1 to n do x[i] := (i-1)/(n-1); od;

                                   x[1] := 0

                                  x[2] := 1/10

                                   x[3] := 1/5

                                  x[4] := 3/10

                                   x[5] := 2/5

                                   x[6] := 1/2
```

$x[7] := 3/5$

$x[8] := 7/10$

$x[9] := 4/5$

$x[10] := 9/10$

$x[11] := 1$

```
>
> print (x);
```

[0, 1/10, 1/5, 3/10, 2/5, 1/2, 3/5, 7/10, 4/5, 9/10, 1]

```
> y:=array(1..n);
```

y := array(1 .. 11, [])

```
> for i from 1 to n do y[i]:=evalf(exp(x[i]));\
> od;
```

$y[1] := 1.$

$y[2] := 1.105170918$

$y[3] := 1.221402758$

$y[4] := 1.349858808$

$y[5] := 1.491824698$

$y[6] := 1.648721271$

$y[7] := 1.822118800$

$y[8] := 2.013752707$

$y[9] := 2.225540928$

$y[10] := 2.459603111$

$y[11] := 2.718281828$

```
> p:=interp(x,y,z);
```

$$p := -.0002865961199\, z^{10} + .001441247795\, z^{9} - .003068783068\, z^{8} + .003928736772\, z^{7} - .001377268517\, z^{6} + .009633790504\, z^{5} + .04128199891\, z^{4} + .1667349377\, z^{3} + .4999935208\, z^{2} + 1.000000243\, z + 1.$$

```
> plot({p(z),E^z},z=1..4);
```

```
> Digits:=40;plot(10^9*(p(z) - E^z),z=0..1);
```

Digits := 40

```
>
> p:=x->a*x^3 + b*x^2+c*x+d;
```

$$p := x \to a\, x^{3} + b\, x^{2} + c\, x + d$$

```
--------------------------------------------------------------------------------
> p(0) =9;

                                     d = 9
--------------------------------------------------------------------------------
> p(0);

                                       d
--------------------------------------------------------------------------------
> q:=unapply(diff(p(x),x),x);

                                          2
                          q := x -> 3 a x  + 2 b x + c
--------------------------------------------------------------------------------
> r:=unapply(diff(q(x),x),x);

                             r := x -> 6 a x + 2 b
--------------------------------------------------------------------------------
> s:=unapply(diff(r(x),x),x);

                                 s := x -> 6 a
--------------------------------------------------------------------------------
> solve({p(0)=1,q(0)=1,r(0)=1,s(0)=1});

                        {c = 1, d = 1, a = 1/6, b = 1/2}
--------------------------------------------------------------------------------
> solve({p(0)=1,q(0)=1,p(1)=E,q(1)});

                    {c = 1, d = 1, b = - 5 + 3 E, a = 3 - 2 E}
--------------------------------------------------------------------------------
> evalf(");

                {c = 1., d = 1., b = 3.154845484, a = -2.436563656}
--------------------------------------------------------------------------------
>
```

project 5 is almost ready.
2 dec 94

Due Date 7 Dec.

Will it be Numerical Integration? yes

For each function f and numbers a < b (f(x), a, b given later)

- For each n = 8, 16, 32, 64, 128, 512
  - Find The T_n, E_n, E~n, CT_n, CE_n, E_n/E_n/2, R_n, RE_n
    1. T_n = Trapezoid rule for n.
    2. E_n = Trap rule error. (I(f) - T_n(f))
    3. E~n = asymptotic estimate.
    4. CT_n = Corrected Trap rule for n.
    5. CE_n = Corrected Trap rule error (I(f) - CT_n(f))
    6. (for n > 8) Error ratio
    7. (for n > 8) R_n = Richardson extrapolation formula
    8. (for n > 8) RE_n = Richardson Error (I(f) - R_n)

  - Find The S_n, E_n, E~n, CS_n, CE_n, E_n/E_n-1, R_n, RE_n
    1. S_n = Simpson rule for n.
    2. E_n = Simpson rule error. (I(f) - S_n(f))
    3. E~n = asymptotic estimate.
    4. CS_n = Corrected Simpson rule for n.
    5. CE_n = Corrected Simpson rule error (I(f) - CS_n(f))
    6. (for n > 8) Error ratio
    7. (for n > 8) R_n = Richardson extrapolation formula
    8. (for n > 8) RE_n = Richardson Error (I(f) - R_n)

Functions f to come, but there are many problems from the book:
7.1 #2; 7.2 #1, 5

You will need maple proc's Trap(f, a, b, n); CTrap(f, a, b, n);
Simpson(f, a, b, n) and CSimpson(f, a, b, n).

The functions f will either be integrable using maples int(f(x),x=a..b);
or the correct integral will be given.

Here are the f, a, and b

1. f(x) = exp (x), a = -1, b = 1. page 189.
2. f(x) = sin (x) * sin(x), a = 0, b = 2*Pi
3. f(x) = exp (-x*x), a = 0, b = 1 page 165.
4. f(x) = exp (cos (x)), a = 0, b = Pi/4. page 171.

MAD 3401 Intro Numerical Math **Test 1** 21 Sep 1994

Show **ALL** work for credit; be neat; and use only **ONE** side of each page of paper.

1. Find the error estimates:
   A. Calculate the error and the relative error, if $x_T = 123.123$ and $x_A = 123.246$.
   B. Suppose x_T and y_T round to $x = 123.4$ and $y = 5.67$, find the smallest interval estimate for the true value of $x - y$.
   C. Suppose x_T rounds to $x_A = 0.987$. Bound the error $\sin(x_T) - \sin(x_A)$.

2. Write the Taylor polynomials with remainder about $x = 0$ for both $f(x) = (1-x)^{-1}$ and $g(x) = (1+x)^{-1}$.

3. The Taylor polynomial with remainder for f is given below (where $c = c_x$ is between 0 and x).

$$f(x) = 1 + 2x^1 + 4x^2 + 8x^3 + \cdots + 2^n x^n + c^2 x^{n+1}$$

   Find the Taylor polynomial with remainder (be sure to include a statement on what c is between) for $xf(x^2)$.

4. Use Taylor polynomial approximations to avoid the loss-of-significance errors in the following expression when x is near 0.

$$\frac{e^x - e^{-x}}{2x}$$

5. For $f(x) = x^5 - 10x^2 - 20$ find the Taylor polynomial with remainder about $x = 2$ for $n = 2$. Bound the error for $1 \le x \le 3$.

6. Use Taylor polynomials with remainder term to evaluate the following limit.

$$\lim_{x->0} \frac{\sin x - xe^x}{x^2}$$

7. Let $p_n(x)$ be the Taylor Polynomial for $f(x) = e^{-x}$ about $x = 0$. How large should n be chosen to have $|f(x) - p_n(x)| \le 10^{-2}$, for $0 \le x \le 2$?

8. Consider the function $f(x)$ defined below. Note that $f(x)$ is piecewise the Taylor polynomial for $\cos(x)$. Show that the error $|\cos(x) - f(x)| < 0.01$ for each x in the domain of $f(x)$.

$$f(x) = \begin{cases} 1 & 0 \le x < 0.1 \\ 1 - \frac{x^2}{2!} & 0.1 \le x < 0.6 \\ 1 - \frac{x^2}{2!} + \frac{x^4}{4!} & 0.6 \le x < 1.3 \end{cases}$$

MAD 3401 Intro Numerical Math **Test 2** 26 Oct 1994

Show **ALL** work for credit; be neat; and use only **ONE** side of each page of paper.

1. For $f(x) = \sin x, a_0 = 1$ and $b_0 = 5$ (thus $c_0 = (1+5)/2$) calculate a_n, b_n and c_n by the bisection method for $n = 1, 2, 3$.

2. For $f(x) = x^2, x_0 = 1$ and $x_1 = \frac{1}{2}$, calculate x_2 and x_3 by the secant method.

3. Derive Newton's method for $f(x)$ yielding x_{n+1} in terms of x_n.

4. Use Newton's method to find the square root of 5. Write a polynomial with **INTEGER** coefficients $f(x)$ so that $\sqrt{5}$ is a root. Write the Newton's formula which yields x_{n+1} in terms of x_n for this particular $f(x)$. Starting from $x_0 = 2$ do two iterations of this formula.

5. For each non-zero constant c, the function $g(x) = cx(1-x)$ has two fixed points, one of which is zero. Find α, the non-zero fixed point of $g(x)$, and determine those values of c for which the iteration $x_{n+1} = g(x_n)$ will converge to α provided x_0 starts close enough to α.

6. Suppose one implements an IEEE style 8 bit floating point binary using *seeeeemm* — that is, 1 sign bit followed by 5 bits for the exponent, stored in excess 15 notation, followed by the mantissa. As in IEEE, the exponent 11111 is only used for infinity and not a number, while the exponent 00000 is only used for zero. Assume $\bar{x}$ is normalized so that $1 \leq \bar{x} < 2$.
   A. How many bits (including an implied bit) are there available for the mantissa?
   B. What floating point number (in decimal) is represented by the bit pattern 11001110?
   C. What is the largest floating point number (in decimal)?
   D. What is the smallest strictly positive floating point number (in decimal)?

7. The initial interval for the bisection method has length $b - a = 2^{-1}$, how many iterations are necessary to get the interval length within 10^{-12}? Suppose the same problem (which has root α,) when using Newton's method, starts with $|\alpha - x_0| = 2^{-1}$, and the estimate $\alpha - x_{n+1} = (\alpha - x_n)^2$ is true. How many iterations of Newtons method are needed to get within 10^{-12}?

8. Suppose $f(x)$ is a fuction with a zero at $x = \alpha$ and $f'(\alpha) \neq 0$. Derive the following error estimate for Newton's method
$$(\alpha - x_{n+1}) = M(\alpha - x_n)^2$$
and find M.

   Hint: Expand $f(x)$ using Taylor's with remainder about $x = x_n$. (The remainder should be the quadratic term.) Substitute $x = \alpha$ and $f(\alpha) = 0$ and use Newton's formula. Don't forget to say what M is.

MAD 3401 Intro Numerical Math **Test 3** 30 Nov 1994

Show **ALL** work for credit; be neat; and use only **ONE** side of each page of paper.

1. Write the Chebyshev polynomials $T_n(x)$ for $n = 0, 1, 2$ and 3 in terms of x.

2. Find the cubic polynomial $p(x)$ which satisfies:
   A. $p(x_0) = p(x_1) = p(x_3) = 0$, but $p(x_2) = 1$.
   B. $p(x_0) = p'(x_0) = p''(x_0) = 0$, but $p(x_1) = 1$.
   C. $p(x_0) = p'(x_0) = p(x_1) = 0$, but $p'(x_1) = 1$.

3. Consider a table of values of $f(x) = \sin x, 0 \leq x \leq 1.6$, with the x entries given in steps of h. Bound the error of linear interpolation in this table when $h = 0.01$.

4. Find the divided differences $D_i = f[x_0, \ldots, x_i]$ for $i = 1, \ldots, 3$ (that is the values $f[x_0, x_1], f[x_0, x_1, x_2]$ and $f[x_0, x_1, x_2, x_3]$) and **USE THESE DIFFERENCES** to find the interpolating polynomials $P_1(x), P_2(x)$ and $P_3(x)$ for the data $(0,0)(1,1)(2,-2)(3,-3)$. (Hint P_1 only uses the first two points, P_2 uses all but the last point.) **SIMPLIFY YOUR POLY'S**

5. Consider the integral $I(f) = \int_{-2}^{2} 2^x dx$.
   A. Find the estimate $T_4(f)$ for $I(f)$.
   B. Find the estimate $S_4(f)$ for $I(f)$.
   C. Find the asymptotic error estimate $\tilde{E}_4^T(f)$ for $I(f)$.

6. Consider the integral $I(f) = \int_0^2 xe^x dx$. Note $e^2 < 8$
   A. Bound the error $E_n^T(f)$ if the trapezoidial rule with $n = 8$ is used to estimate $I(f)$.
   B. Bound the error $E_n^S(f)$ if Simpson's rule with $n = 8$ is used to estimate $I(f)$.

7. A minimax polynomial $p(x)$ of degree 7 is use to approximate $\ln x$ for $1 \leq x \leq 2$. Estimate the maximum error $|\ln x - p(x)|$. The estimates $2 < e < 3$ and $2^{10} \doteq 10^3$ may be used.

8. If $z = (x_0 + x_1)/2$ and $2h = x_1 - x_0$, show $f[x_0, x_1] - f'(z) \doteq h^2 f'''(z)/6$. Hints: Let z be the midpoint and let $z + h = x_1, z - h = x_0$; and expand f in a Taylors series for $f(z \pm h)$ about z.

The list of formulas to be put on the board:

$$\frac{(x-x_0)(x-x_1)\cdots(x-x_n)}{(n+1)!}f^{(n+1)}(c)$$

$$h[\frac{1}{2}f(x_0)+f(x_1)+f(x_2)+\cdots+f(x_{n-1})+\frac{1}{2}f(x_n)]$$

$$\frac{h}{3}[f(x_0)+4f(x_1)+2f(x_2)+4f(x_3)+\cdots+4f(x_{n-1})+f(x_n)]$$

$$\frac{[(b-a)/2]^{n+1}}{2^n(n+1)!}\max_{a\le x\le b}|f^{(n+1)}(x)|$$

$$\frac{-h^2(b-a)f''(c)}{12}$$

$$\frac{-h^2(f'(b)-f'(a))}{12}$$

$$\frac{-h^4(b-a)f^{(4)}(c)}{180}$$

$$\frac{-h^4(f'''(b)-f'''(a))}{180}$$

$$f[x_0,x_1,\ldots x_n]\doteq\frac{f^n(c)}{n!}$$

$$T_{n+1}=2xT_n-T_{n-1}, n\ge 1$$

$$R_{2n}=\frac{2^pI_{2n}-I_n}{2^p-1}; I-I_n\doteq\frac{c}{n^p}$$

$$P_{k+1}(x)=P_k(x)+(x-x_0)\cdots(x-x_k)f[x_0,\ldots,x_{k+1}]$$

$$\sum_{i=1}^{n}w_if(x_i)$$

$$x_{n+1}=x_n-f(x_n)\cdot\frac{x_n-x_{n-1}}{f(x_n)-f(x_{n-1})}$$

$$bn+m\sum x_i=\sum y_i; b\sum x_i+m\sum x_i^2=\sum x_iy_i$$

```
> with(stats):U:=RandUniform(0..1):n:=20;x := array(1..n):

                                    n := 20
```

```
> for i from 1 to n do
> x[i]:=U():
> od:print (x);

[ .7974857856, .4450269247, .6386273485, .0460241146, .2057229320, .2320193122,

    .8146898380, .8553867838, .8046682595, .5479921284, .0557140299,

    .1198106710, .3609305579, .9712272244, .5805483826, .8735338387,

    .3489794196, .4912196910, .7216131848, .5391035580 ]
```

```
> for i from 1 to n do x[i] := evalf(Pi*U() + Pi): od:print ( x);

[ 4.124313925, 5.410274189, 5.411567868, 4.754821254, 3.891563121, 3.537946168,

    4.661464691, 4.003075614, 3.995747827, 4.477830840, 5.279190466,

    5.688831830, 3.822096011, 5.876164918, 3.365368962, 4.030508253,

    3.507286797, 4.019845552, 4.484473867, 6.002731688 ]
```

```
>
```

```
> f:=x->x^3;

                                                 3
                                   f := x -> x
--------------------------------------------------------------------------------
> diff(f(x),x);

                                            2
                                       3 x
--------------------------------------------------------------------------------
> g:=unapply(",x);

                                                 2
                                  g := x -> 3 x
--------------------------------------------------------------------------------
> (y-f(3) )= g(3) * (x - 3);

                               y - 27 = 27 x - 81
--------------------------------------------------------------------------------
> solve(",y);

                                   - 54 + 27 x
--------------------------------------------------------------------------------
> h:=unapply("",x);

                              h := x -> - 54 + 27 x
> ;
--------------------------------------------------------------------------------
> plot({f(x), g(x), h(x)}, x=0..4);
--------------------------------------------------------------------------------
>
```

```
> readlib(spline);

proc(X,Y,z,d) ... end

--------------------------------------------------------------------------------
> F:='spline/makeproc'(spline([-1,0,1],[0,1,0],x,3), x);

F := proc(x) if x < 0 then 1-3/2*x^2-1/2*x^3 else 1-3/2*x^2+1/2*x^3 fi end

--------------------------------------------------------------------------------
> G:='spline/makeproc'(spline([-1,-1/2,0,1/2,1],[0,0,1,0,0],x,3), x);

G := proc(x)
         if x < -1/2 then 18/7+66/7*x+72/7*x^2+24/7*x^3
         elif x < 0 then 1-60/7*x^2-64/7*x^3
         elif x < 1/2 then 1-60/7*x^2+64/7*x^3
         else 18/7-66/7*x+72/7*x^2-24/7*x^3
         fi
     end

--------------------------------------------------------------------------------
> H:='spline/makeproc'(spline([-1,-2/3,-1/3,0,1/3,2/3,1],[0,0,0,1,0,0,0],x,3), x);

H := proc(x)
         if x < -2/3 then -36/13-9*x-243/26*x^2-81/26*x^3
         elif x < -1/3 then 36/13+207/13*x+729/26*x^2+405/26*x^3
         elif x < 0 then 1-513/26*x^2-837/26*x^3
         elif x < 1/3 then 1-513/26*x^2+837/26*x^3
         elif x < 2/3 then 36/13-207/13*x+729/26*x^2-405/26*x^3
         else -36/13+9*x-243/26*x^2+81/26*x^3
         fi
     end

--------------------------------------------------------------------------------
> K:='spline/makeproc'(spline([-1,-3/4,-1/2,-1/4,0,1/4,1/2,3/4,1],[0,0,0,0,1,0,0,0,0],x,3), x

K := proc(x)
         if x < -3/4 then 180/97+564/97*x+576/97*x^2+192/97*x^3
         elif x < -1/2 then -306/97-1380/97*x-2016/97*x^2-960/97*x^3
         elif x < -1/4 then 270/97+2076/97*x+4896/97*x^2+3648/97*x^3
         elif x < 0 then 1-3408/97*x^2-7424/97*x^3
         elif x < 1/4 then 1-3408/97*x^2+7424/97*x^3
         elif x < 1/2 then 270/97-2076/97*x+4896/97*x^2-3648/97*x^3
         elif x < 3/4 then -306/97+1380/97*x-2016/97*x^2+960/97*x^3
         else 180/97-564/97*x+576/97*x^2-192/97*x^3
         fi
     end

--------------------------------------------------------------------------------
> plot({'F(x)','G(x)','H(x)','K(x)'},x=-1..1);
--------------------------------------------------------------------------------
>
```

```
> T:=proc(n, f, a, b)
> local sum, h, i;
> sum:=0;
> h := evalf((b-a)/n);
> for i from 0 to n do
> sum := sum + evalf(f(a + i*h));
> od;
> sum := sum - evalf((f(a) + f(b))/2);
> sum := h*sum;
> end;

T := proc(n,f,a,b)
    local sum,h,i;
        sum := 0;
        h := evalf((b-a)/n);
        for i from 0 to n do  sum := sum+evalf(f(a+i*h)) od;
        sum := sum-evalf(1/2*f(a)+1/2*f(b));
        sum := h*sum
    end
```

```
> n := 2;f:=x->E^x;Digits:=8;
```

n := 2

$$f := x \to E^x$$

Digits := 8

```
> while ( n < 1024 ) do
> T(n, f, 0, 1);
> n := 2 * n;
> od;
```

1.7539311

n := 4

1.7272219

n := 8

1.7205186

n := 16

1.7188411

n := 32

1.7184217

n := 64

1.7183167

n := 128

1.7182909

n := 256

1.7182842

n := 512

```
                                 1.7182825

                                 n := 1024
--------------------------------------------------------------------------------
> E
> ;

                                     E
--------------------------------------------------------------------------------
> evalf(E);

                                 2.7182818
--------------------------------------------------------------------------------
>
```