# Gragg Extrapolation Method

```
> restart;Digits:=20;
```

$$Digits := 20$$

```
> libname:="c:/nalib",libname;
```

$libname$ := "/nalib", "/Library/Frameworks/Maple.framework/Versions/14/lib", "/Library/Frameworks/Maple.framework/Versions/14/toolbox/NAG/lib"

```
> with(numanal);
```

[*SOR*, *SOR_dir*, *adaptq*, *adaptq_dir*, *bezier*, *bezier_dir*, *bisection*, *bisection_dir*, *chop*, *chop_dir*, *clamped_spline*, *clamped_spline_dir*, *divided_diff*, *divided_diff_dir*, *extrap*, *extrap_dir*, *falseposition*, *falseposition_dir*, *fixedpoint*, *fixedpoint_dir*, *gaussseidel*, *gaussseidel_dir*, *hermite*, *hermite_dd*, *hermite_dd_dir*, *hermite_dir*, *horner*, *horner_dir*, *jacobi*, *jacobi_dir*, *muller*, *muller_dir*, *natural_spline*, *natural_spline_dir*, *newton*, *newton_dir*, *romberg*, *romberg_dir*, *secant*, *secant_dir*, *steffensen*, *steffensen_dir*]

We examine the Gragg extrapolation method on the IVP $\frac{\partial}{\partial t} y = y - t^2 + 1, y(0) = 0.5$ on [0,2], $h = 0.1$.

```
> deq:=diff(y(t),t)=y(t)-t^2+1;
```

$$deq := \frac{d}{dt} y(t) = y(t) - t^2 + 1$$

```
> init:=y(0)=.5;
```

$$init := y(0) = 0.5$$

We first find the exact solution, which we rewrite as a function $Y$.

```
> soln:=dsolve({deq,init},y(t));
```

$$soln := y(t) = 1 + 2t + t^2 - \frac{1}{2} e^t$$

```
> Y:=unapply(rhs(soln),t);
```

$$Y := t \rightarrow 1 + 2t + t^2 - \frac{1}{2} e^t$$

We get the directions for the procedure **extrap**.

```
> extrap_dir();
extrap returns a table of values for the ODE.

t[i] is the t value, w[i] the approximation at y(t[i]),

h[i] the stepsize, and k is the level of extrapolation.

The arguments for extrap are:
(1)function expression in t and y
(2)lefthand endpoint
(3)righthand endpoint
(4)initial value
(5)tolerance
(6)stepsize
```

We extract the right-hand side of the differential equation and replace each occurrance of $y(t)$ by $y$.

```
> f:=rhs(deq);
```

$$f := y(t) - t^2 + 1$$

```
> f:=subs(y(t)=y,f);
```

$$f := y - t^2 + 1$$

We execute the procedure **extrap**.

```
> extrap(f,0,2,0.5,10^(-10),.1,.01);
GRAGG EXTRAPOLATION
```

| t[i] | w[i] | h[i] | K |
|---|---|---|---|
| 0.1000 | 0.6574145410 | 0.1000 | 4 |
| 0.2000 | 0.8292986209 | 0.1000 | 4 |
| 0.3000 | 1.0150705962 | 0.1000 | 4 |
| 0.4000 | 1.2140876512 | 0.1000 | 4 |
| 0.5000 | 1.4256393646 | 0.1000 | 4 |
| 0.6000 | 1.6489405998 | 0.1000 | 4 |
| 0.7000 | 1.8831236463 | 0.1000 | 4 |
| 0.8000 | 2.1272295358 | 0.1000 | 4 |
| 0.9000 | 2.3801984444 | 0.1000 | 4 |
| 1.0000 | 2.6408590858 | 0.1000 | 4 |
| 1.1000 | 2.9079169880 | 0.1000 | 4 |
| 1.2000 | 3.1799415386 | 0.1000 | 4 |
| 1.3000 | 3.4553516662 | 0.1000 | 4 |
| 1.4000 | 3.7324000166 | 0.1000 | 4 |
| 1.5000 | 4.0091554648 | 0.1000 | 4 |
| 1.6000 | 4.2834837878 | 0.1000 | 4 |
| 1.7000 | 4.5530263041 | 0.1000 | 4 |
| 1.8000 | 4.8151762678 | 0.1000 | 4 |
| 1.9000 | 5.0670527789 | 0.1000 | 4 |
| 2.0000 | 5.3054719505 | 0.1000 | 4 |

Finally, we check a couple of values.

```
> Y(1.0);
```

2.6408590857704773823

```
> Y(2.0);
```

5.3054719505346748864