```
# Prof. Dr. Serkan Dağ
# ME 310 Numerical Methods
# File 13.3
# Ralston's Method
# Solves the first-order ODE dy/dt = f(t,y)
# Applicable for both linear and nonlinear ODE's
# Interval of interest is [a,b]
# n is the number of segments
# y0 is the initial condition
```

```
restart :
Digits := 16 :
```

```
# Open the output file
```

```
out := fopen("E:\\Courses\\2019_Spring\\ME310\\Files\\result2.txt", 'WRITE', 'TEXT') :
```

```
# Define the right-hand-side function
```

```
f := -k·(y - ya) :
```

```
k := 0.25 :
ya := 20 :
```

```
# Interval end points
```

```
a := 0. :
b := 5.015 :
```

```
# Number of segments
```

```
n := 16 :
```

```
# Initial condition
```

```
y0 := 90. :
```

```
# Step size
```

```
h := (b - a)/n;
```

$$h := 0.3134375000000000 \tag{1}$$

```
# Initiate the integration
```

```
t0 := a :
for i from 1 by 1 to n
 while true do

 if i = 1 then
printf("\n %5.1f %15.10f %8.4f", i - 1, y0, t0);
fprintf(out, "\n %5.1f %15.10f %8.4f", i - 1, y0, t0) :
 end if:

k1 := subs({t = t0, y = y0}, f) :
k2 := subs({t = t0 + 3·h/4, y = y0 + 3·k1·h/4}, f) :

yralston := y0 + (k1/3 + 2·k2/3)·h :

tn := t0 + h :

printf("\n %5.1f %15.10f %8.4f", i, yralston, tn);
 fprintf(out, "\n %5.1f %15.10f %8.4f", i, yralston, tn) :
```

```
    t0 := tn :
    y0 := yralston :

    end do:


    0.0    90.0000000000    0.0000
    1.0    84.7297504578    0.3134
    2.0    79.8562942046    0.6269
    3.0    75.3497569599    0.9403
    4.0    71.1825136559    1.2538
    5.0    67.3290190964    1.5672
    6.0    63.7656513646    1.8806
    7.0    60.4705670207    2.1941
    8.0    57.4235672019    2.5075
    9.0    54.6059738031    2.8209
   10.0    52.0005149804    3.1344
   11.0    49.5912192743    3.4478
   12.0    47.3633177052    3.7612
   13.0    45.3031532393    4.0747
   14.0    43.3980970711    4.3881
   15.0    41.6364712086    4.7016
   16.0    40.0074768874    5.0150
> fclose(out) :
>
```