Exercises from Sec.8.3

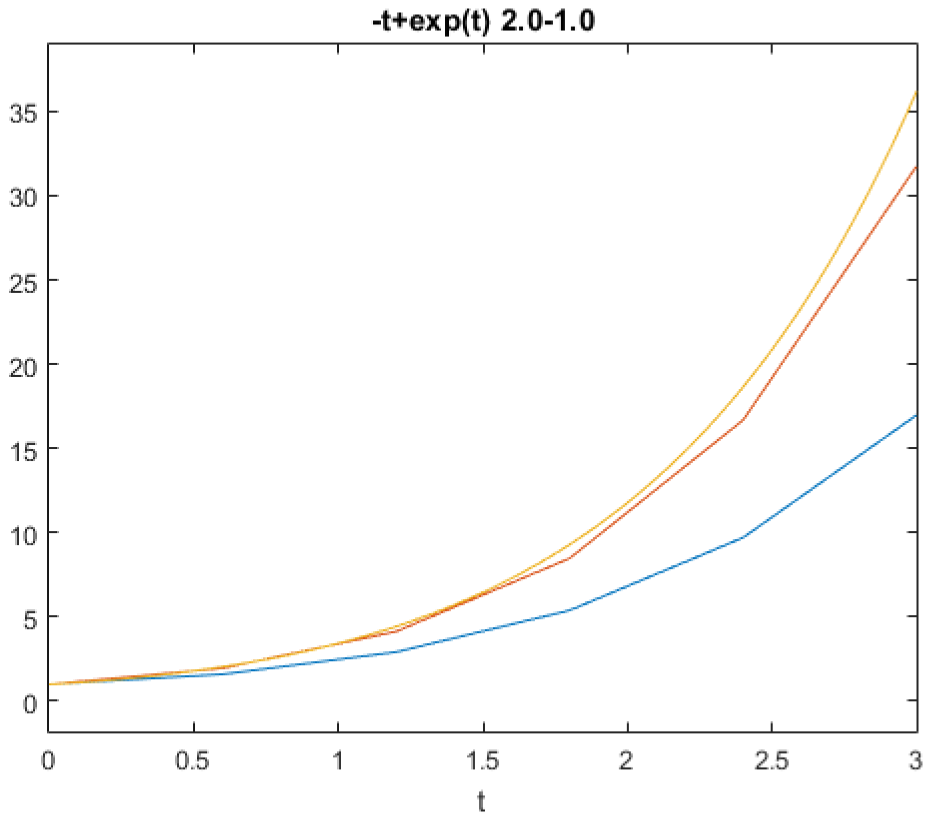
Table of Contents

: 20141510 , ## :

1.1

```
clc
clear
syms fEuler1 fEuler2 dfdt1 dfdt2

tI = 0;
tF = 3;
dt = 3/5;
fInitialValue = 1;
times = tI:dt:tF;

fEuler1(1) = fInitialValue;  % first-order Euler method
fEuler2(1) = fInitialValue;  % second-order Euler method

for i = 1:1:(length(times)-1)
    dfdt1(i) = times(i) + fEuler1(i);
    fEuler1(i+1) = fEuler1(i) + dfdt1(i) * dt;
end

for i = 1:1:(length(times)-1)
    dfdt1(i) = times(i) + fEuler2(i);
    dfdt2(i) = 1 + dfdt1(i);
    fEuler2(i+1) = fEuler2(i) + dfdt1(i) * dt + (1/2)*dfdt2(i)*(dt^2);
end

double(fEuler1(length(times)))  % first order approximation of f(3)
double(fEuler2(length(times)))  % second order approximation of f(3)
g = dsolve('Df=t+f','f(0)=1'); % solve the diff-equ using dsolve
f = inline(g);
f(3)

plot(times,fEuler1)
hold on
plot(times,fEuler2)
ezplot(f,times)
hold off
```

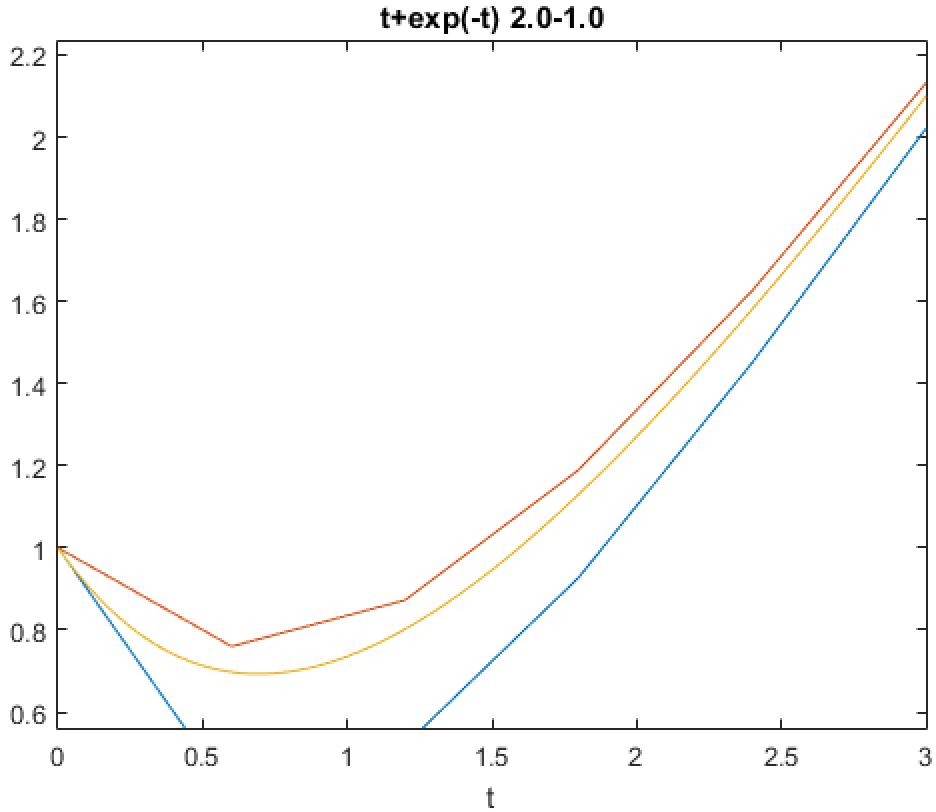
```
ans =

   16.9715


ans =

   31.7380


ans =

   36.1711
```



1.2

```
clc
clear
syms fEuler1 fEuler2 dfdt1 dfdt2

tI = 0;
tF = 3;
dt = 3/5;
fInitialValue = 1;
times = tI:dt:tF;
```

```
fEuler1(1) = fInitialValue;  % first-order Euler method
fEuler2(1) = fInitialValue;  % second-order Euler method

for i = 1:1:(length(times)-1)
    dfdt1(i) = times(i) - fEuler1(i);
    fEuler1(i+1) = fEuler1(i) + dfdt1(i) * dt;
end

for i = 1:1:(length(times)-1)
    dfdt1(i) = times(i) - fEuler2(i);
    dfdt2(i) = 1 - dfdt1(i);
    fEuler2(i+1) = fEuler2(i) + dfdt1(i) * dt + (1/2)*dfdt2(i)*(dt^2);
end

double(fEuler1(length(times)))  % first order approximation of f(3)
double(fEuler2(length(times)))  % second order approximation of f(3)
g = dsolve('Df=t-f','f(0)=1'); % solve the diff-equ using dsolve
f = inline(g);
f(3)

plot(times,fEuler1)
hold on
plot(times,fEuler2)
ezplot(f,times)
hold off


ans =

    2.0205


ans =

    2.1313


ans =

    2.0996
```

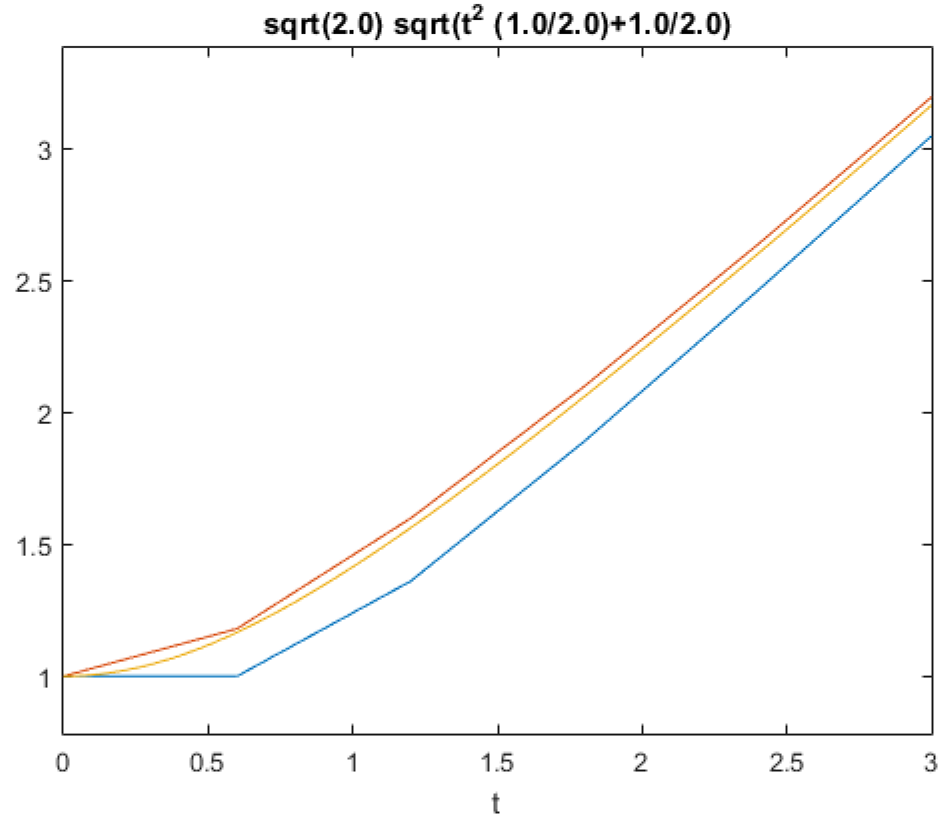


1.3

```
clc
clear
syms fEuler1 fEuler2 dfdt1 dfdt2

tI = 0;
tF = 3;
dt = 3/5;
fInitialValue = 1;
times = tI:dt:tF;

fEuler1(1) = fInitialValue;  % first-order Euler method
fEuler2(1) = fInitialValue;  % second-order Euler method

for i = 1:1:(length(times)-1)
    dfdt1(i) = times(i) / fEuler1(i);
    fEuler1(i+1) = fEuler1(i) + dfdt1(i) * dt;
end

for i = 1:1:(length(times)-1)
    dfdt1(i) = times(i) / fEuler2(i);
    dfdt2(i) = (fEuler2(i)-times(i)*dfdt1(i))/(fEuler2(i)^2);
    fEuler2(i+1) = fEuler2(i) + dfdt1(i) * dt + (1/2)*dfdt2(i)*(dt^2);
end
```

```
double(fEuler1(length(times)))  % first order approximation of f(3)
double(fEuler2(length(times)))  % second order approximation of f(3)
g = dsolve('Df=t/f','f(0)=1'); % solve the diff-equ using dsolve
f = inline(g);
f(3)

plot(times,fEuler1)
hold on
plot(times,fEuler2)
ezplot(f,times)
hold off
```

```
ans =

    3.0461


ans =

    3.1934


ans =

    3.1623
```

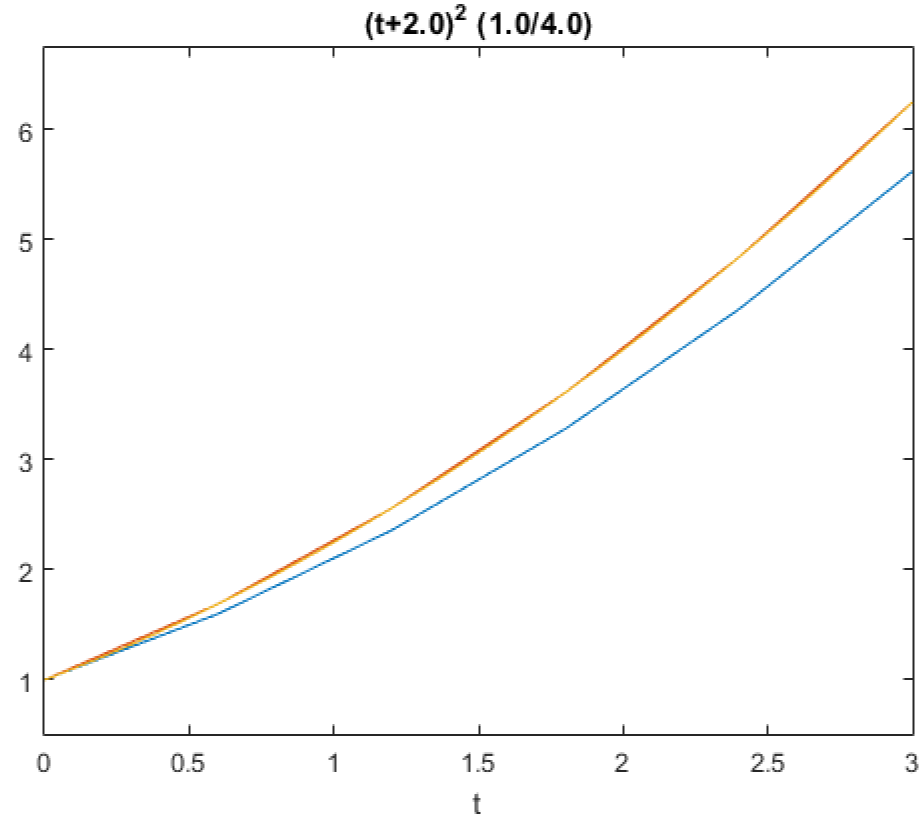


1.4

```
clc
clear
syms fEuler1 fEuler2 dfdt1 dfdt2

tI = 0;
tF = 3;
dt = 3/5;
fInitialValue = 1;
times = tI:dt:tF;

fEuler1(1) = fInitialValue;  % first-order Euler method
fEuler2(1) = fInitialValue;  % second-order Euler method

for i = 1:1:(length(times)-1)
    dfdt1(i) = fEuler1(i)^(1/2);
    fEuler1(i+1) = fEuler1(i) + dfdt1(i) * dt;
end

for i = 1:1:(length(times)-1)
    dfdt1(i) = fEuler2(i)^(1/2);
    dfdt2(i) = (1/2) * fEuler2(i)^(-1/2) * dfdt1(i);
    fEuler2(i+1) = fEuler2(i) + dfdt1(i) * dt + (1/2)*dfdt2(i)*(dt^2);
end
```

```
double(fEuler1(length(times)))  % first order approximation of f(3)
double(fEuler2(length(times)))  % second order approximation of f(3)
g = dsolve('Df=f^(1/2)','f(0)=1'); % solve the diff-equ using dsolve
f = inline(g(1));
f(3)

plot(times,fEuler1)
hold on
plot(times,fEuler2)
ezplot(f,times)
hold off
```

```
ans =

    5.6211


ans =

    6.2500


ans =

    6.2500
```



Published with MATLAB® R2015a