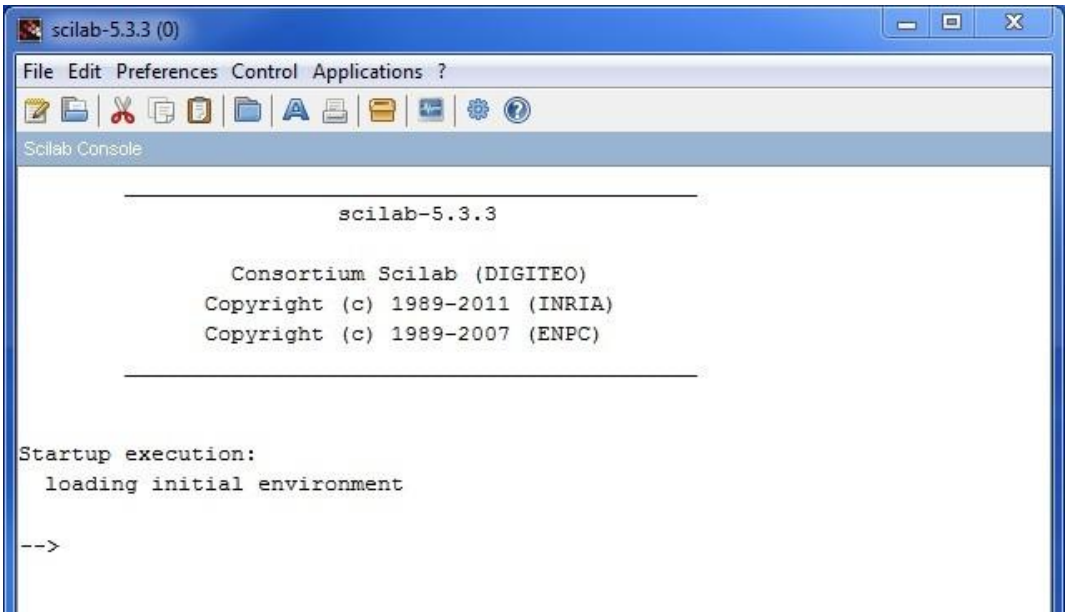
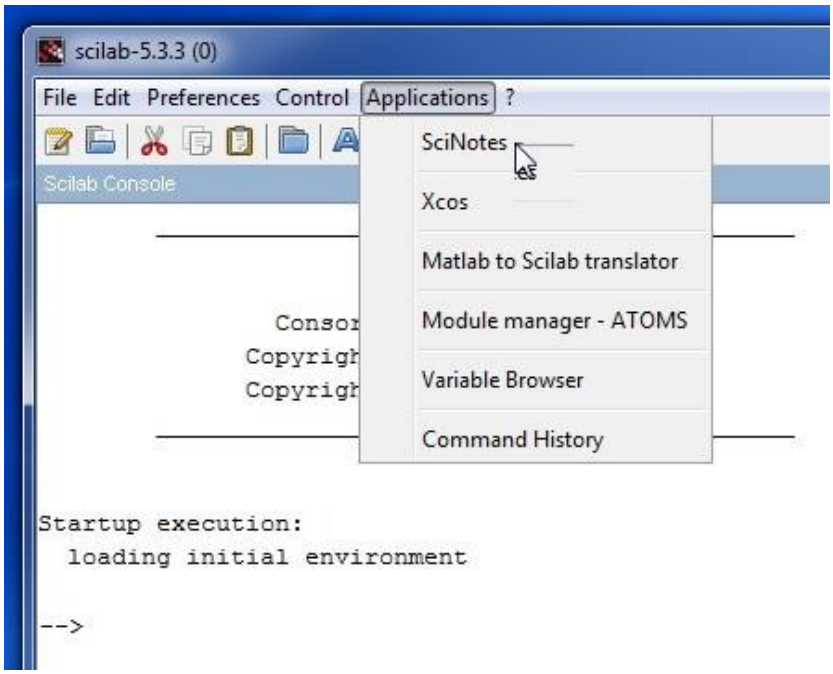
Széchenyi István Egyetem
Műszaki Tudományi Kar

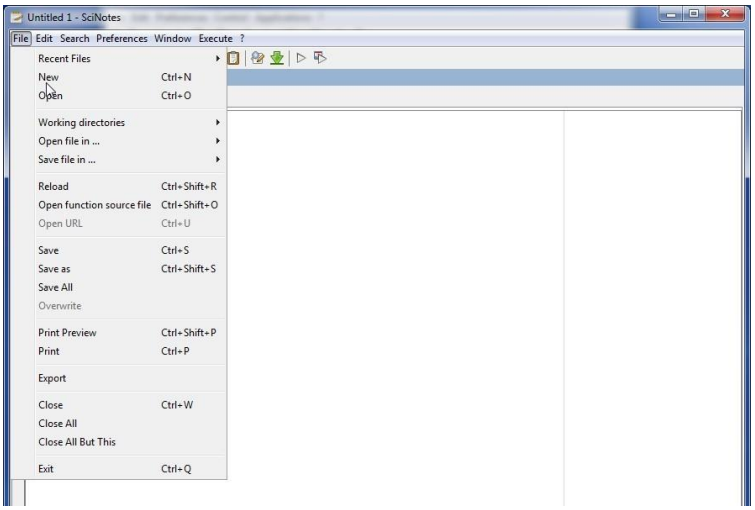
Alkalmazott Mechanika
Tanszék

GÉPEK DINAMIKÁJA 2.gyak.hét – 1. és 2. Feladat

(kidolgozta: Dr. Nagy Zoltán egyetemi adjunktus)







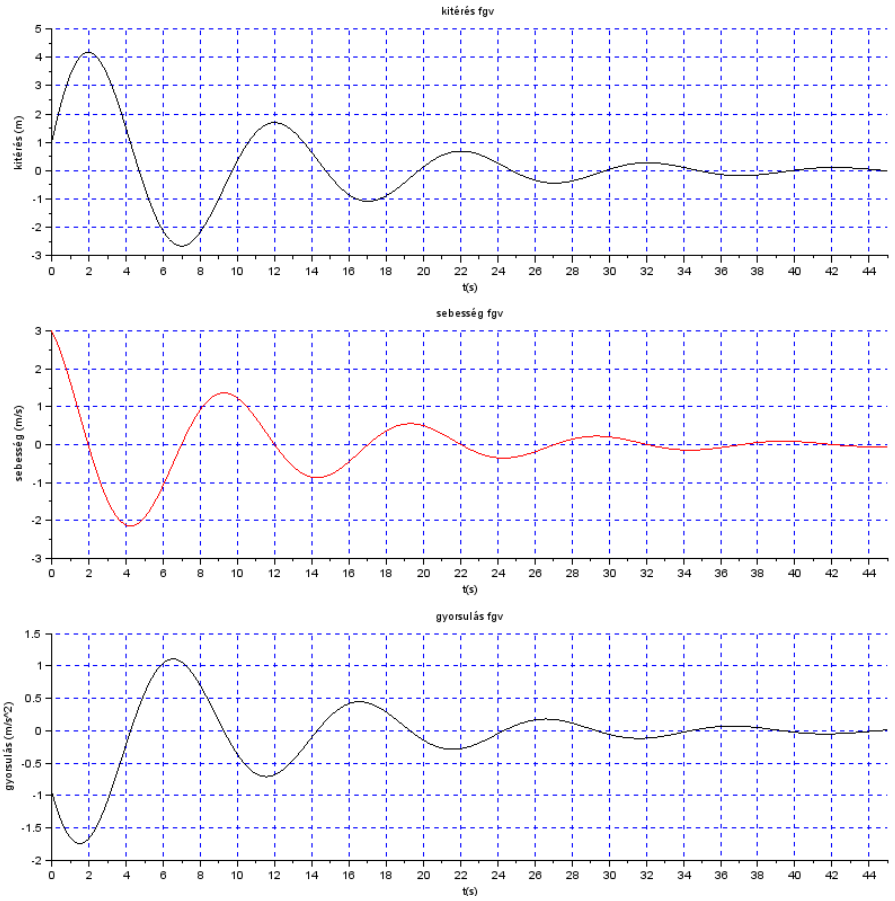
2.gyak.hét – 1. feladat

Program begépelése:

```
// 2.hét - 1.feladat
clear all;
usecanvas(%T);
        // Másodrendű állandó együtthatós differenciálegyenlet megoldása
        // Runge-Kutta módszerrel:  5y''+0.9y'+2y=0 .
        // Átrendezés:  y''=-(0.9/5)y'-(2/5)y  .... ezt kell megoldanunk.
        // A megoldás kezdeti feltételei:  t = 0 (s) ➔ y(0)=1 (m) elmozdulás,
        //                                 t = 0 (s) ➔ y'(0)=3 (m/s) sebesség.
        //-----------------------------------------------------------------------------------------
        // legyen a megoldás kételemű oszlopvektora: p, ahol
        //                         p(1)=y (m) elmozdulás,
        //                         p(2)=y' (m/s) sebesség.
        // ennek derivált vektora: p' , ahol p'(1)=y'  azaz  p(2) (m/s) sebesség,
        //                 p'(2)=y'' (m/s^2) gyorsulás,
        //                 p'(2)=y''=-(0.9/5)*y'-(2/5)*y.
        //                 p'(2)=-(0.9/5)*p(2)-(2/5)*p(1).
        //-----------------------------------------------------------------------------------------
function [pdot]=f(t, p)
        // a p' , azaz a derivált vektor megadása következik, amit majd integrálunk !!!
pdot(1)=p(2)                    // sebesség, ezt mindig így, ilyen formában kell megadni !!
pdot(2)=-(0.9/5)*p(2)-(2/5)*p(1)        // a fgv. gyorsulásra kifejezett alakja
endfunction
        // Diff. egyenlet kezdeti feltételeinek megadása az előírások alapján
t0=0;                   // kezdeti időpont zérus
p0(1)=1;                // kezdeti elmozdulás , azaz y(0)=1 (m)
p0(2)=3;                // kezdeti sebesség, azaz y'(0)= 3 (m/s).
        // Megoldás időintervalluma
t=0:0.002:45.0;         // 45 másodperc, 0,002 -es lépésközzel végrehajtva
        // Runge-Kutta módszer aktivizálása
p=ode("rk",p0,t0,t,f); // rk - a megoldó módszer jelen esetben Runge-Kutta, p0 - a kezdeti
                        értékek vektora, t0 - a kezdeti időpont, t - azaz idő melyre a megoldást
                        meghatározzuk, f - a jobb oldala a differenciál egyenletnek.
```

```
        ////////////////////////  VIGYÁZAT !!!  ////////////////////
        // Ezzel a módszerrel a gyorsulás függvényt nem tudjuk kiíratni, hiszen
        // a megoldást jelentő p vektor első eleme az elmozdulás, a második eleme
        // pedig a sebesség értékeit tartalmazza! Amennyiben a gyorsulást is
        // szeretnénk kiíratni, így ezt újból fel kell írni a programban!
a=-(0.9/5)*p(2,:)-(2/5)*p(1,:);          //  gyorsulás fgv (m/s^2)
        // Eredmények kirajzoltatása
subplot(3,1,1)
        //A subplot paranccsal a grafikus képernyőt tudjuk felosztani. Az első két szám jelenti,
        hogy egy 3x1-es mátrixra osztjuk a képernyőt az utolsó szám pedig, hogy a mátrixon
        belül hol jelenítsük meg az adott grafikont.
plot2d(t,p(1,:),1);
xtitle(" kitérés fgv","t(s)","kitérés (m)"); // az első paraméter a grafikon felirat, a második az
                                              x tengely felirat, a harmadik az y tengely felirat
xgrid(2);            // a grafikon mögé egy rácsot definiálhatunk ezzel a paranccsal
subplot(3,1,2)
plot2d(t,p(2,:),5);
xtitle("sebesség fgv","t(s)","sebesség (m/s)");
xgrid(2);
subplot(3,1,3)
plot2d(t,a(1,:),1);
xtitle("gyorsulás fgv","t(s)","gyorsulás (m/s^2)");
xgrid(2);
```

FUTTATÁS: Execute >>>> .... file with no echo



>>> A program mentése valamilyen néven

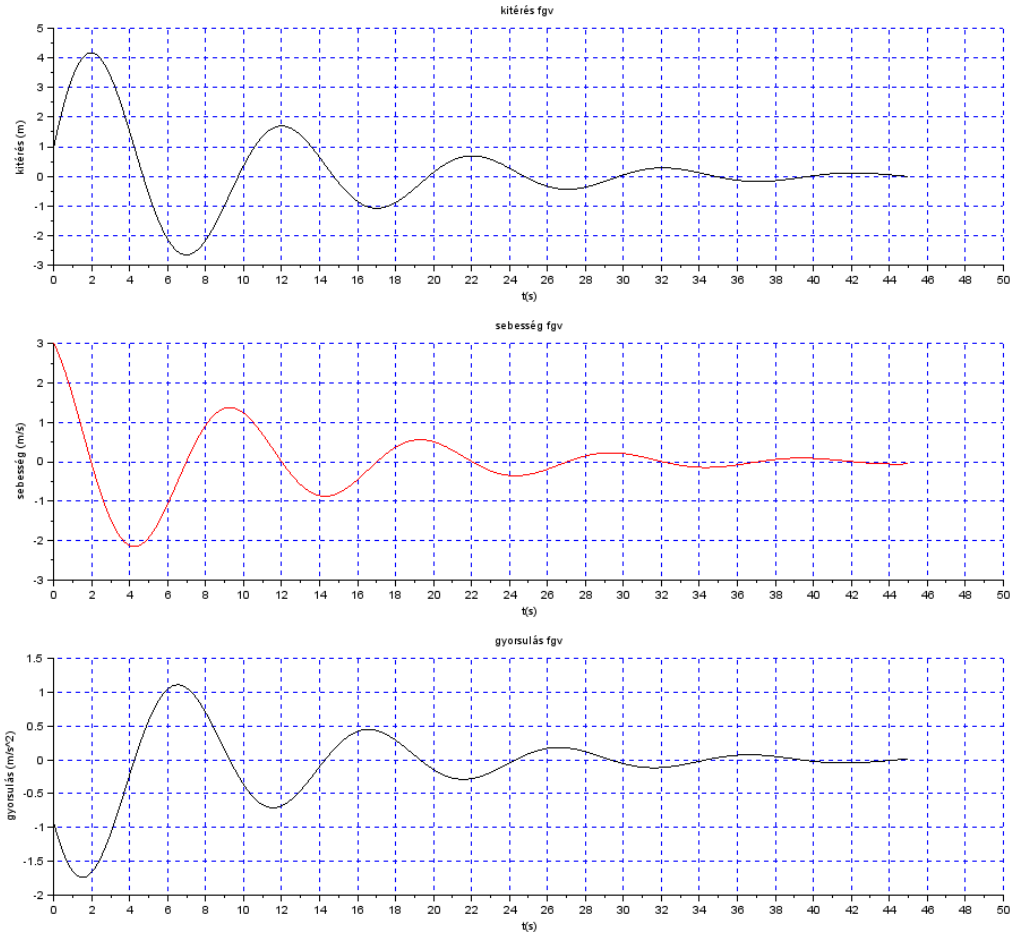
2.gyak.hét – 2. feladat

Program begépelése:

```
// 2.hét - 2.feladat
clear all;
usecanvas(%T);
        // Másodrendű állandó együtthatós differenciálegyenlet megoldása
        // Euler módszerrel:  5y''+0.9y'+2y=0 .
        // Átrendezés:  y''=-(0.9/5)y'-(2/5)y  .... ezt kell megoldanunk.
        // A megoldás kezdeti feltételei:  t = 0 (s) ➔  y(0)=1 (m) elmozdulás,
        //                                 t = 0 (s) ➔  y'(0)=3 (m/s) sebesség.
        //-----------------------------------------------------------------------------------------------
        //  Megoldás időintervalluma
dt=0.002;
tmax=45.0;
n=int(tmax/dt);
        //  Diff. egyenlet kezdeti feltételeinek megadása az előírások alapján
t0=0;                           // kezdeti időpont zérus (s).
y0=1.0;                         // kezdeti elmozdulás, azaz y(0)=1 (m).
v0=3.0;                         // kezdeti sebesség, azaz y'(0)= 3 (m/s).
a0=-(0.9/5)*v0-(2/5)*y0;        // kezdeti gyorsulás: ezt számítani kell
        ////////////////////
t0=dt;
t=(1:n);
y=(1:n);
v=(1:n);
a=(1:n);
        // indítás pillanatában   //////  i=1 ciklusváltozó ////////////
t(1)=0;
y(1)=y0;
v(1)=v0;
a(1)=a0;
        /////////// for ciklus kezdete ////////////////////////////////
for i=2:n
    t(i)=t0+dt;
    a(i)=-(0.9/5)*v0-(2/5)*y0;
    v(i)=v0+((a(i)+a0)/2)*dt;
    y(i)=y0+((v(i)+v0)/2)*dt;
        /// A következő, új ciklus kezdő értékei (értékcsere)
    t0=t(i);
    y0=y(i);
    v0=v(i);
    a0=a(i);
end
        //  Eredmények kirajzoltatása
subplot(3,1,1)
plot2d(t,y(1,:),1);
xtitle("kitérés fgv","t(s)","kitérés (m)");
xgrid(2);
```

```
subplot(3,1,2)
plot2d(t,v(1,:),5);
xtitle("sebesség fgv","t(s)","sebesség (m/s)");
xgrid(2);
subplot(3,1,3)
plot2d(t,a(1,:),1);
xtitle("gyorsulás fgv","t(s)","gyorsulás (m/s^2)");
xgrid(2);
```

FUTTATÁS: Execute >>>> .... file with no echo



>>> A program mentése valamilyen néven