Problem 1. Write down the equation for position of an object moving horizontally with a constant velocity "v". Assume v=50 m/s, use the Euler method (finite difference) to solve the equation as a function of time.

- Compare your computational result with the exact solution.
- Compare the result for different values of the time-step.

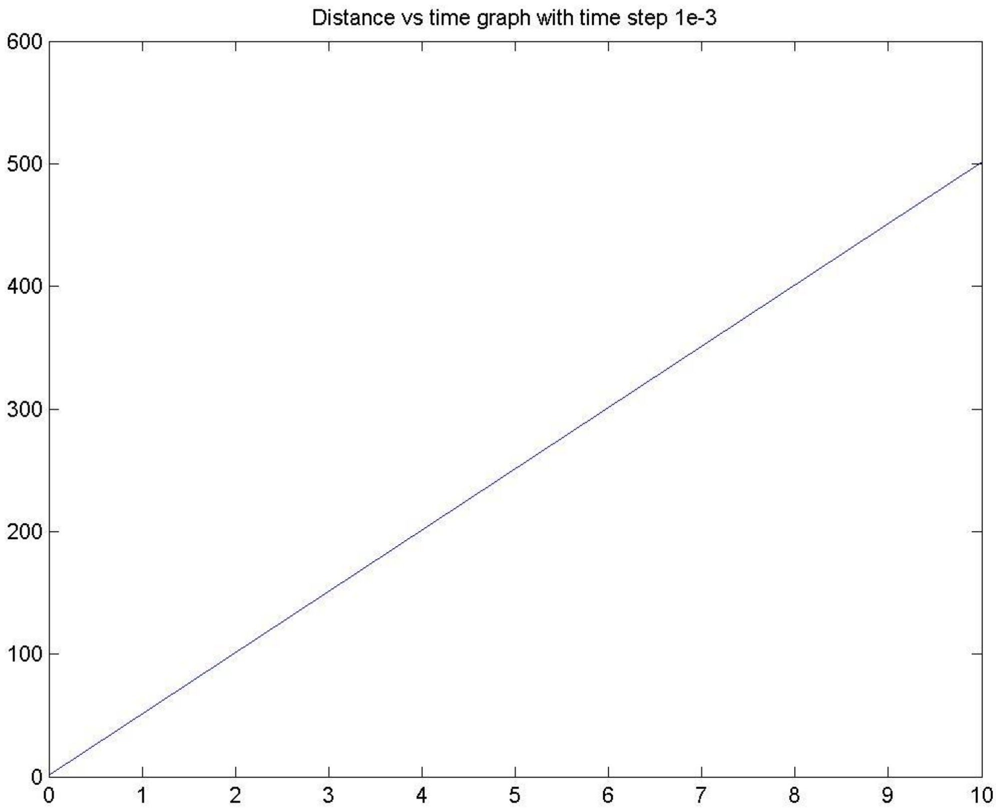
➔ In both methods analytical and computational the graph is same i.e. straight line. The reason for equality is that Euler method and analytical method yield same expression. Here we are considering only the ideal case without any friction. So it has to be straight line.

➔ Even if we change the time steps there is no change. Because there is linear relation between distance and time. So it does not matter if we take so many points in a straight line or only two points. It is sufficient. If it was other than straight line then it can be different.

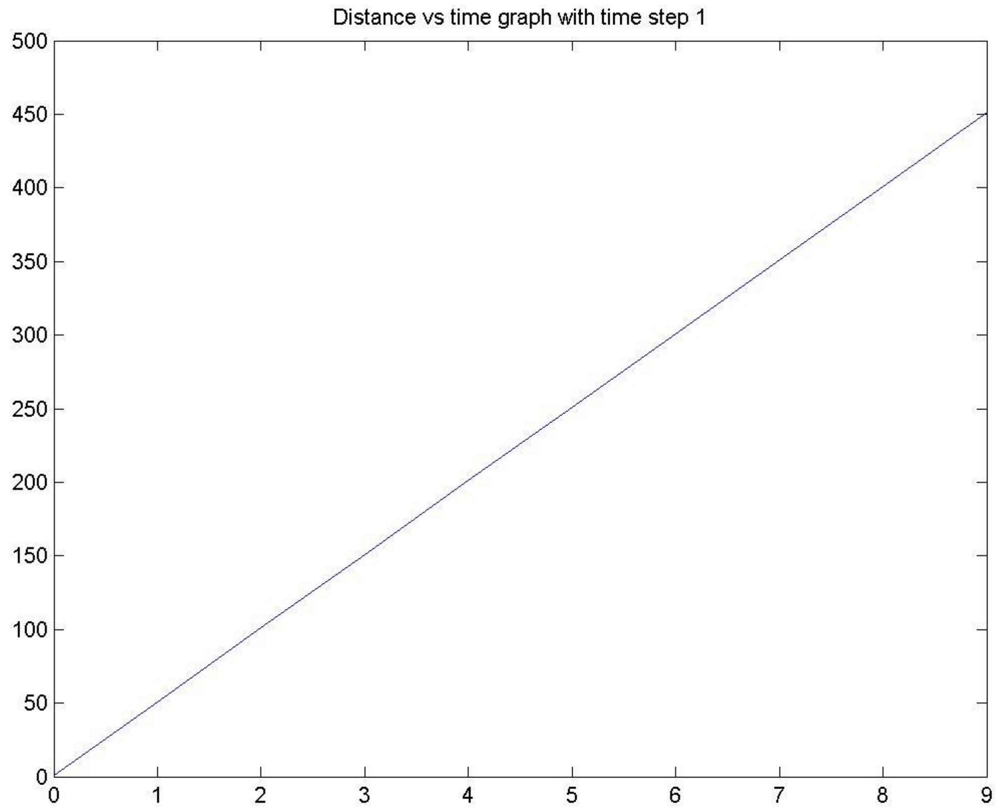
Code :

```
clearall;
tim = 10;
init_dis = 1;
dt = 5;
niter=tim/dt;
time=zeros(niter,1);
dis=zeros(niter,1);
time(1)=0;
dis(1)=init_dis;
vel=50;

for step=1  : niter-1
        dis(step+1)=dis(step)+ vel*dt;
        time(step+1)=time(step)+dt;
end
plot(time, dis);
title('Distance vs time graph with time step 1')
```

Time-step = 0.001



Time-Step=1



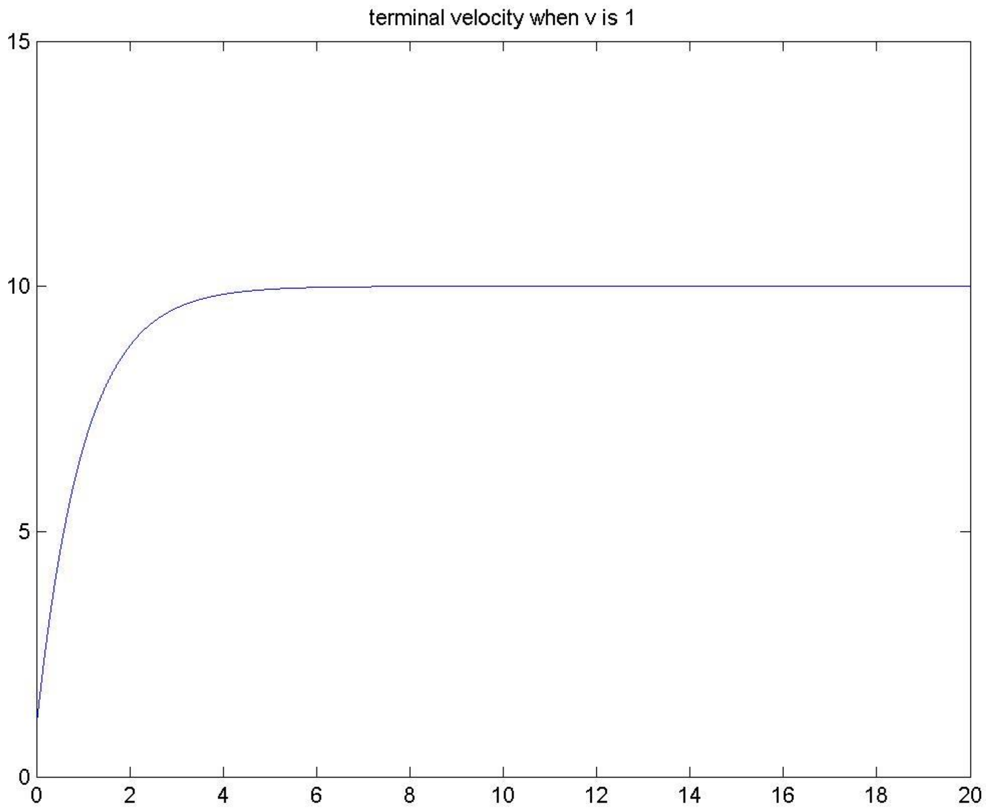
Problem 2: Parachute problem: frictional force on the object increases as the objects moves faster (as we learned today in the class). Role of parachute is to produce the frictional force in the form of air drag. Consider the most simple form, so the equation for velocity: $dv/dt = a - bv$. Where a (from applied force), b (from friction) are constants. Use Euler's method to solve for "v" as a function of time. Choose a=10 and b=1. What is the terminal velocity in this case?

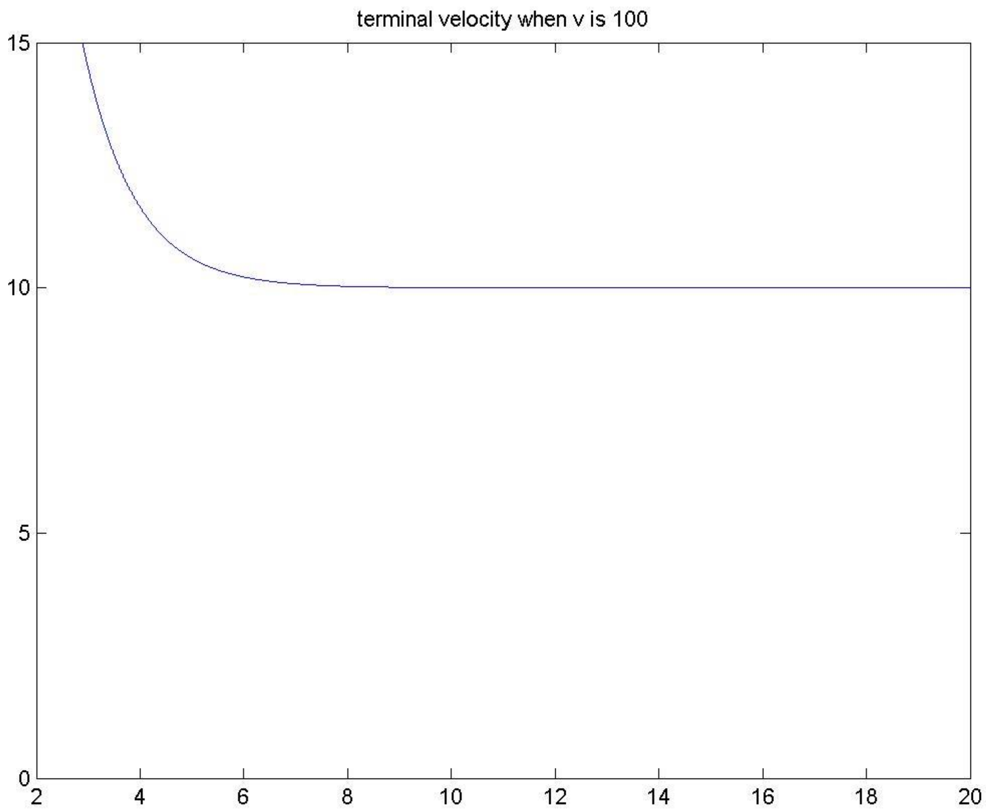
Terminal velocity is 10m/s in this case.

Code :

```
clearall;
tim = 20;
init_vel = 1; %even if we take initial vel more i.e. terminal vel is same though graph is diff.
dt = 1e-3;
niter=tim/dt;
time=zeros(niter,1);
vel=zeros(niter,1);
time(1)=0;
vel(1)=init_vel;
a=10;
b=1;

for step=1  : niter-1
vel(step+1)=vel(step)+ a*dt- b*vel(step)*dt;
time(step+1)=time(step)+dt;
end
plot(time, vel);
ylim([0 15]);
title('terminal velocity')
```

Here if we take initial velocity more than terminal velocity, graph would be decreasing and saturate at terminal velocity. But if we take initial velocity less than terminal velocity then graph would be increasing first and saturate at terminal velocity.


terminal velocity when v is 1
15
10
5
0
0
2
4
6
8
10
12
14
16
18
20


terminal velocity when v is 100
15
10
5
0
2
4
6
8
10
12
14
16
18
20

Problem 3: Population growth problem can be modeled using a rate equation:dN/dt=aN-bN^2

(N) number of individuals which varies with time.

First term (aN) birth of new members

Second term (bN^2) corresponds to death; proportional to N2

Because food will become harder to find when population becomes very large.Use the Euler method to solve the equation as discussed in the class forthe decay problem. Take a=10 and b=0; then take a=10, b=3.Compare your numerical solution with exact solutions.For different values of “a” and “b”, give some explanations regardingyour result.

Code:

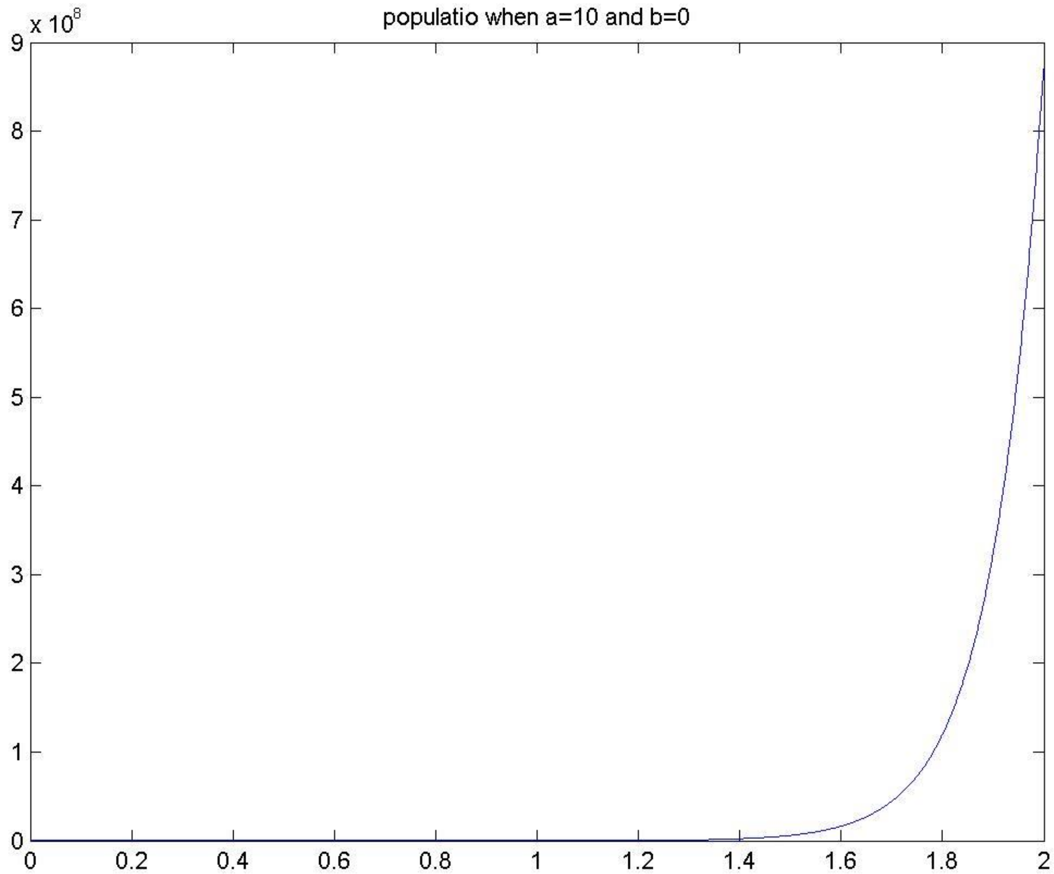
```
clearall;
tim = 2;
init_N = 2;
dt = .001;
niter=tim/dt;
time=zeros(niter,1);
N=zeros(niter,1);
time(1)=0;
N(1)=init_N;
a=10;
b=3;

for step=1 : niter-1
N(step+1)=N(step)+ a*N(step)*dt- b*N(step)*N(step)*dt;
time(step+1)=time(step)+dt;
end
plot(time, N);
xlim([0 2]);
title('populatio when a=10 and b=0')
```

Case-1:

When A=10 and B=0:

No, person will die.

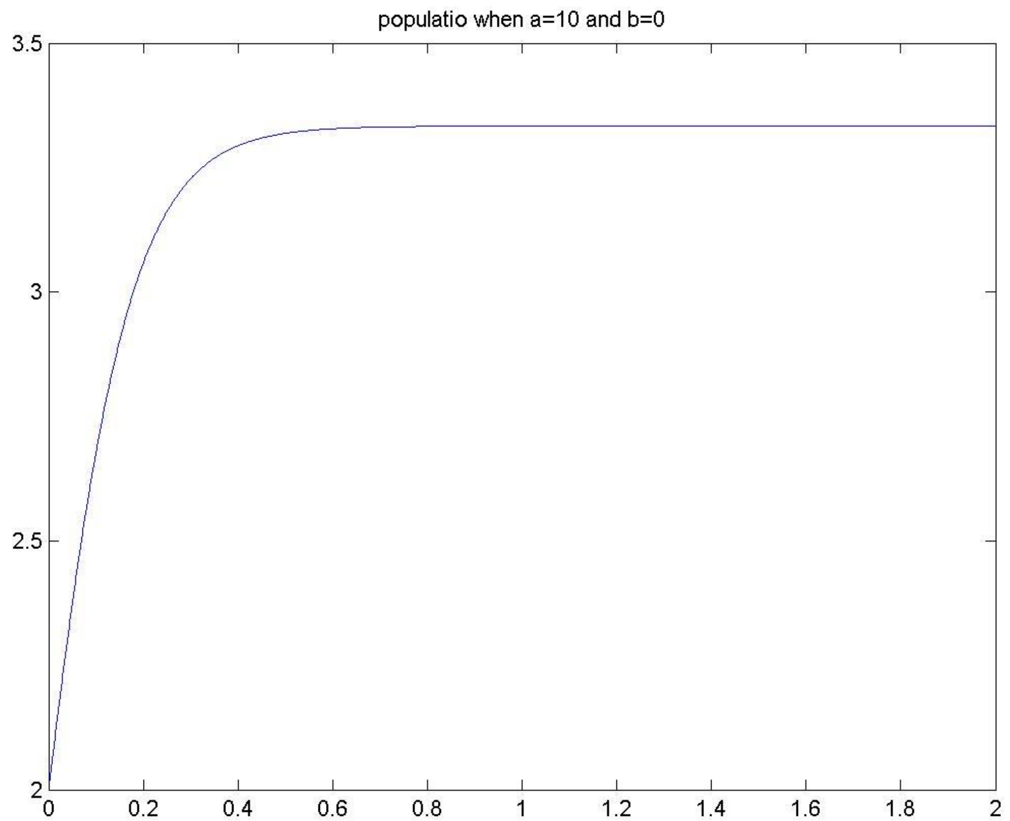
The curve is exponential. So population will grow very faster.



Case-2:

When A=10 and B=3:

Now, when number of people are greater than 4 then it rate of change of population with respect to time will be negative. That shows there will no increase in population beyond certain limit.



Here we can say that for lower values of N first term would dominate (hence graph would increase initially) and for higher values of N, second term would dominate (hence graph would no more be increasing).

Broadly we can say that when the rate of change in population with time is positive than the population will increase. Here we can conclude that when it is zero population will be steady at that point and there will be no increment in population number.

Hence, when the value of N will be A/B there will be no further increment and it is a steady point. After a reasonable amount of time the population number will remain constant and its value is A/B;

Problem-4:

4. Bicycle problem:

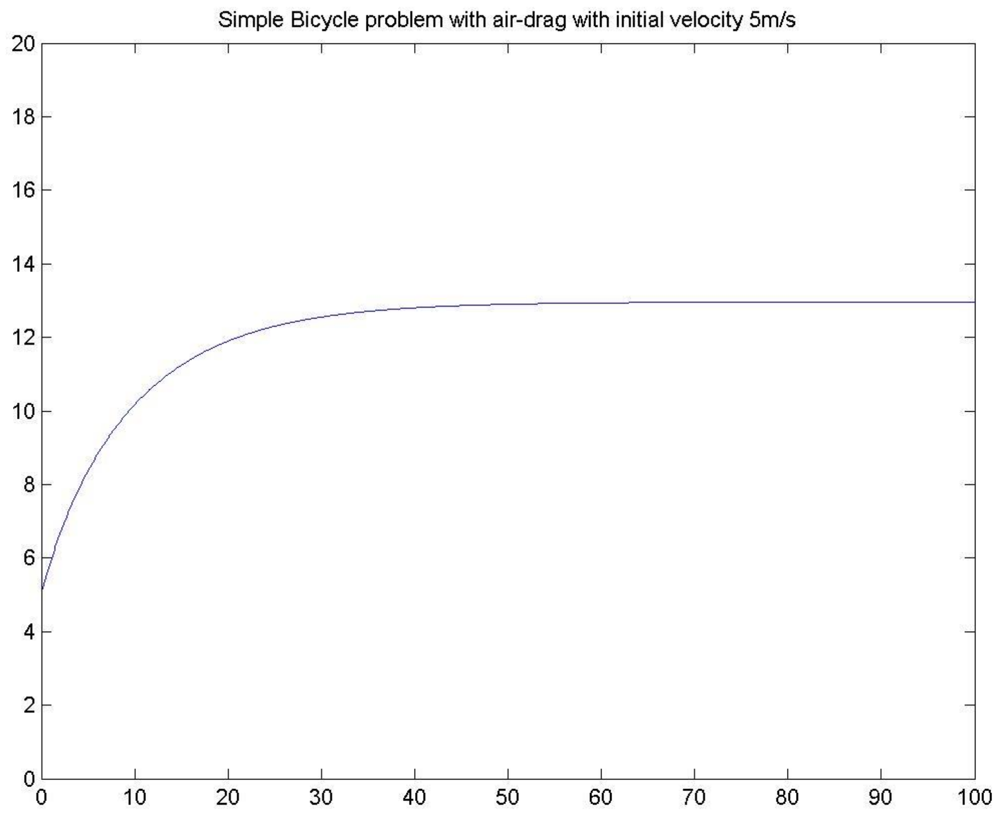
(a) Rewrite the bicycle problem/code as discussed in the class.Investigate the effect of rider's power, mass and frontal area on the ultimate velocity. Generally for a rider in the middle of a group the effective frontal area isabout 30% less than the rider at the front. How much less energy does arider in the group expend than one at the front (assuming both movingat 12.5 m/s).

Code:

```
clear all;
tim = 100;
init_vel=5;
dt =.1;
niter=tim/dt;
time=zeros(niter,1);
vel=zeros(niter,1);
time(1)=0;
vel(1)=init_vel;
mass=75;
power=400;
c=0.5;
area=0.3;
density=1.225;
```

```
for step=1  : niter-1
   vel(step+1)=vel(step)+power*dt/(vel(step)*mass)-(c*density*vel(step)*vel(step)*area*dt)/mass;
time(step+1)=time(step)+dt;
end
plot(time, vel);
ylim([0 20])
title('Simple Bicycle problem with air-drag with initial velocity 5m/s')
```

Graph:



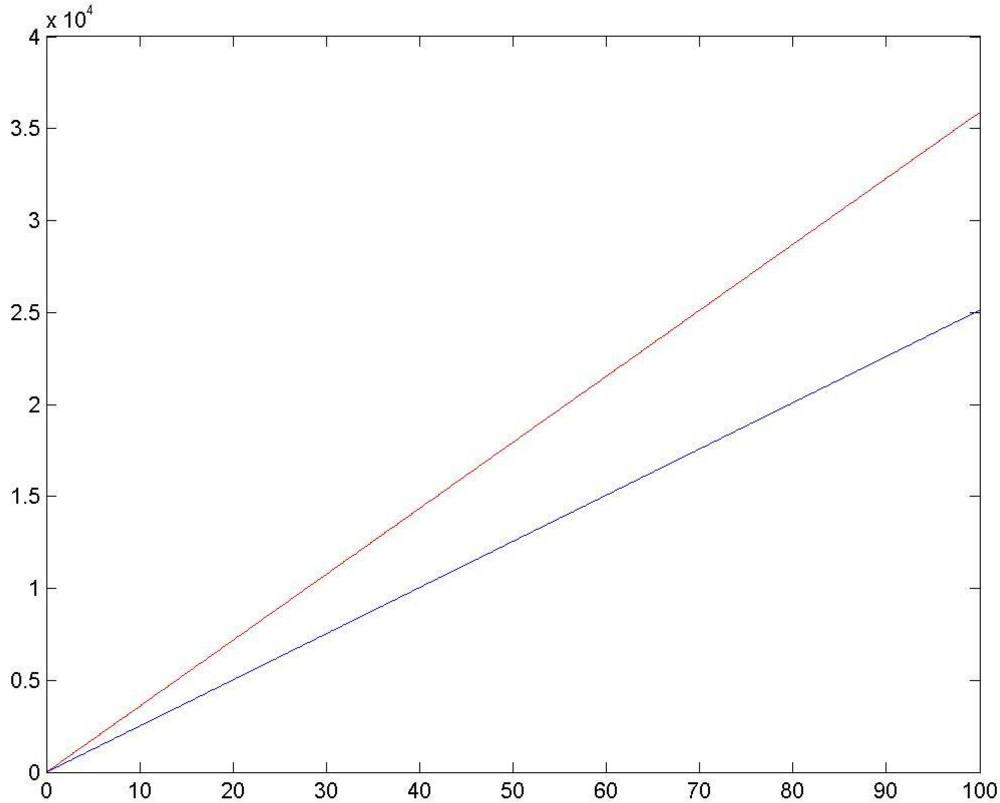
Here all the energy is expended by the bicyclist. The bicyclist at the front will have to expand more energy than the cyclist at the back.

Here both the cyclist move at a constant speed so the only difference b/w the work done is the drag force energy that one has to experience to move with that speed. Now the frontal area of the second bicyclist is 30% less.

Code:

```
clear all;
tim = 100;
init_ene=0;
dt =.1;
niter=tim/dt;
time=zeros(niter,1);
energy1=zeros(niter,1);
energy2=zeros(niter,1);
time(1)=0;
energy1(1)=init_ene;
energy2(1)=init_ene;
mass=75;
power=400;
c=0.5;
area=0.3;
density=1.225;
vel=12.5;
for step=1  : niter-1

energy1(step+1)=energy1(step)+c*density*area*vel*vel*vel*dt;

energy2(step+1)=energy2(step)+c*density*area*vel*vel*vel*dt*0.
7;
    time(step+1)=time(step)+dt;
end
plot(time, energy1,'r');
hold on
plot(time, energy2);
```

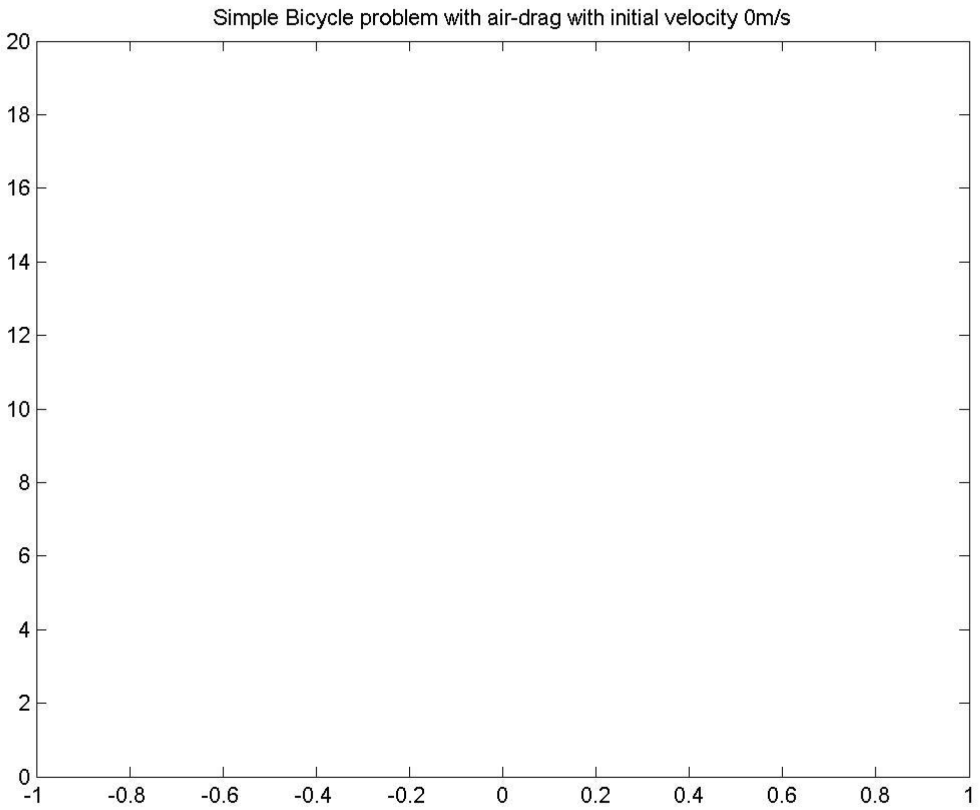
Graph: The red graph is of the bicyclist at front.
And the blue line is of the bicyclist in middle of the group.



(b) Run your code (case (a) discussed during class) with initial v=0;observe the output and give possible explanation. Explain why itis important to give a non-zero initial velocity.

→ Because if we start from rest. The first term which contains 1/v will become infinite. So our equation will break down. So the term dv/dt becomes infinite which is difficult to handle in computational approach. But if we see it physically it doesn't make sense either. We assumed that the bicyclist exerts constant amount of power. And instantaneous power is the product of force and velocity. So if the velocity is zero the force term has to be infinite which physically impossible.

Graph:



(c) As discussed in the class, we have assumed that the bicyclistmaintains a constant power. What about the assumption when thebicycle has a very small velocity? (Instantaneous power=productof force and velocity).

As stated in the previous question we know that for zero velocity assuming the power to a constant quantity is irrelevant. At low velocities it is more realistic to assume, that the rider is able to exert a constant force. That means for small "v" there is a constant force,
So, if we take force to be constant. So, now the equation is:

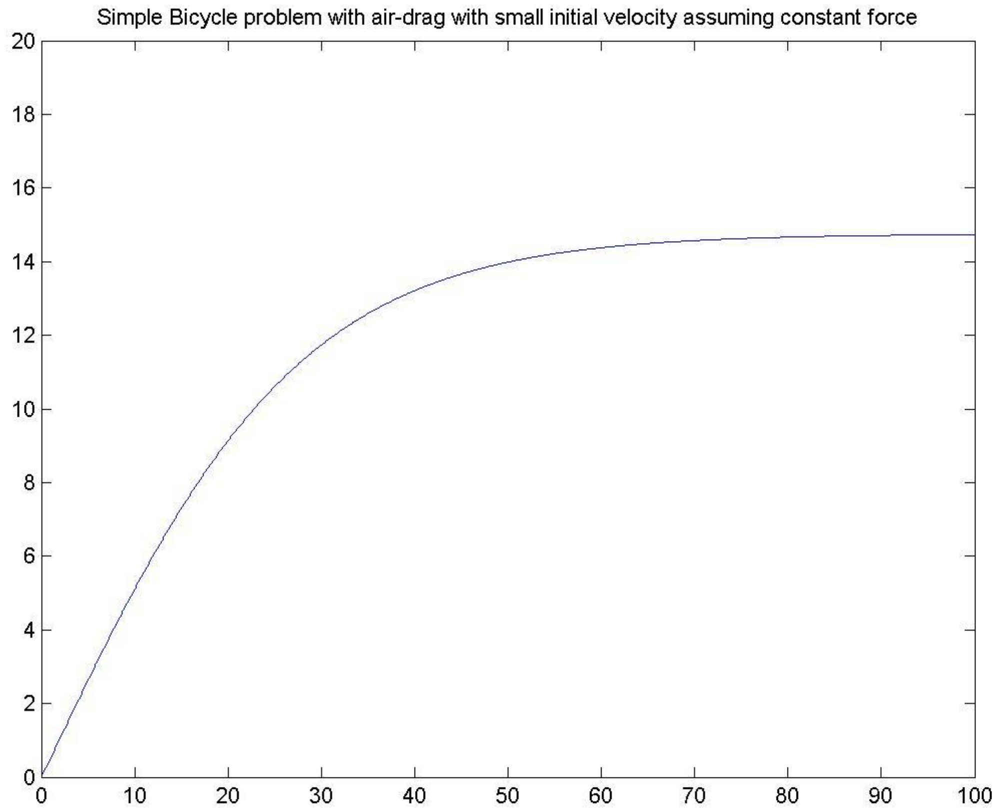
$$dv/dt=Fo/M.$$

Code:

```
clearall;
tim = 100;
init_vel=0.005;
dt =.1;
niter=tim/dt;
```

```
time=zeros(niter,1);
vel=zeros(niter,1);
time(1)=0;
vel(1)=init_vel;
mass=75;
force=40;
c=0.5;
area=0.3;
density=1.225;

for step=1  : niter-1
vel(step+1)=vel(step)+force*dt/mass-
(c*density*vel(step)*vel(step)*area*dt)/mass;
time(step+1)=time(step)+dt;
end
plot(time, vel);
ylim([0 20])
title('Simple Bicycle problem with air-drag with small initial velocity
assuming constant force')
```

Graph:



(d) At low velocities it is more realistic to assume, that the rider isable to exert a constant force. That means for small “v” there is aconstant force, which means eqn is dv/dt=F0/mModify your matlab code to include this term for small velocities,that means we have 2 regimes and 2 eqns one for small velocitiesand one for larger velocities. Make your code work automaticallyfor both the regimes and crossover from small to large v occurwhen the power reaches P(=F0v) . Take F0=P/v where v=5m/s.Change different parameters and report about importantobservations.

Code:

```
clearall;
tim = 100;
init_vel=0.005;
dt =.1;
niter=tim/dt;
time=zeros(niter,1);
vel=zeros(niter,1);
time(1)=0;
vel(1)=init_vel;
mass=75;
force=40;
power=400;
c=0.5;
area=0.3;
density=1.225;

for step=1  : niter-1
ifvel(step)<=5
vel(step+1)=vel(step)+force*dt/mass-
(c*density*vel(step)*vel(step)*area*dt)/mass;
end
ifvel(step)>5
```

```
        vel(step+1)=vel(step)+power*dt/(vel(step)*mass)-
(c*density*vel(step)*vel(step)*area*dt)/mass;
end
time(step+1)=time(step)+dt;
end
plot(time, vel);
ylim([0 20])
title('Simple Bicycle problem with air-drag with 2 regimes and
2 equations')
```

Graph: