Assignment-6

Problem -1:

Code:

```
clear all;
init_x=-10;
fin_x=10;
dx=0.1;
init_y=-10;
fin_y=10;
dy=0.1;
niter=(fin_x-init_x)/dx;
[x,y] = meshgrid(-10:1:10,   -10:1:10);
u=0.2.*x.*x+0.5.*y.*y+20;
v=-0.1.*y.*y.*y+0.5.*x.*x-10;
figure
quiver(x,y,u,v)
 starty=-10:1:10;
startx = -10*ones(size(starty));
%starty =  -9.9:1:10;
streamline(x,y,u,v,startx,starty)
```

A ->What physical insight do you get from the plot?

---> Here we can see that at any given point There is only one way where the object can go, this means this streamline by definition do not intersect. Streamlines provide a image at any given instance of this wind velocity characteristics. We can know the history and the flow of the object.

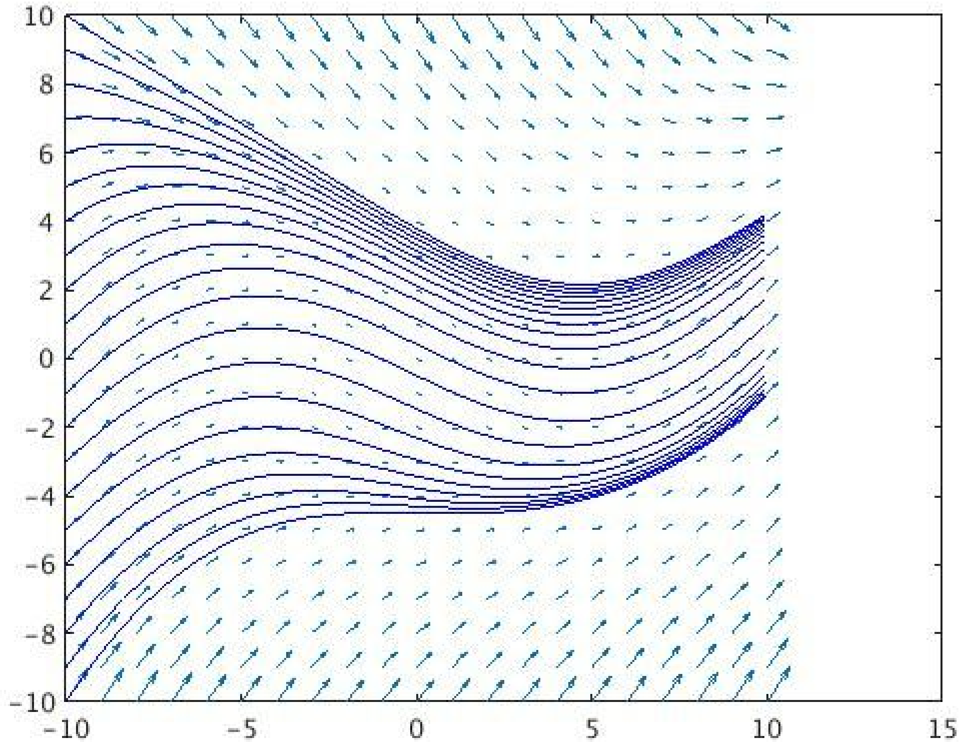
B) What does arrows that you produced with the quiver command show at each point?

---> This quiver plot consists of velocity and position vectors. And the first arguments are of position vectors,So this quiver plot shows velocity vectors as arrows with components (vx,vy) at each point (x,y).

C:) What does streamline shows?

Streamline are curves that are instantaneously tangent to the velocity vector(vx,vy) of the flow at any given point at any given time. Streamline shows that if we put a object at any given point and provided that no external forces are acting here then it will travel in the direction of the streamline at any point in time.

Graph: Streamline for Wind velocity:



Problem : 2

Code:

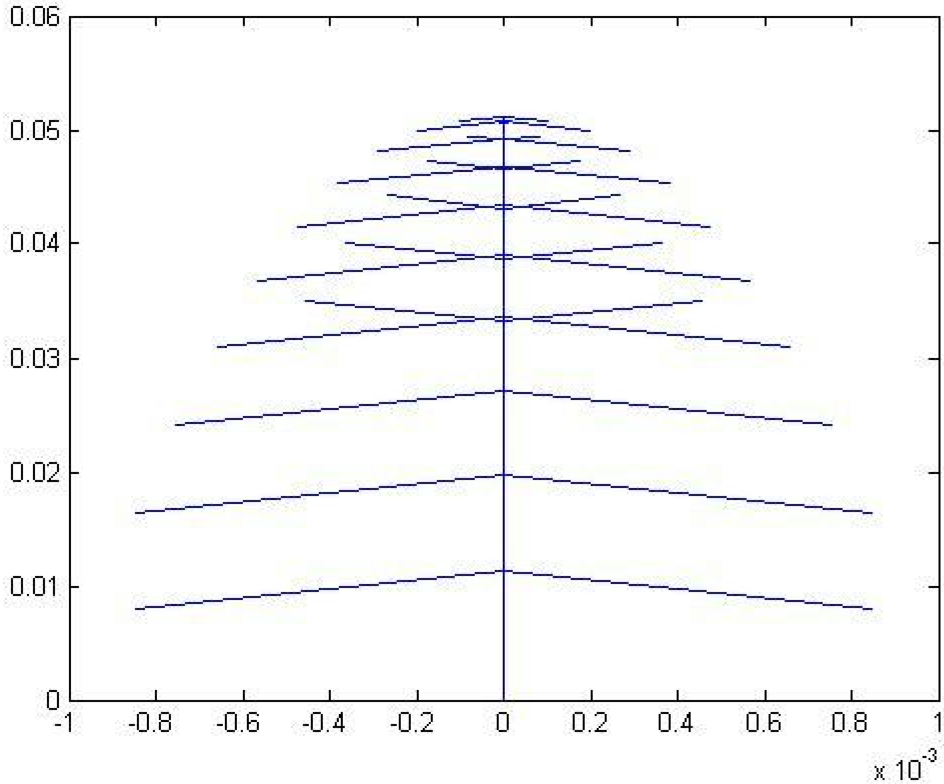
```
clear;close all;
timescale=30;
dt=1;
tstart=0;
tfinal=timescale;
niter=tfinal/dt;
u0=zeros(2,1);
u0(1)=0; % initial position
u0(2)=200; % initial velocity
[t,u]=ode45(@rhs,[tstart:dt:tfinal],u0);
x=u(:,1);
v=u(:,2);
figure
quiver(zeros(size(x)),x,zeros(size(x)),v);
```

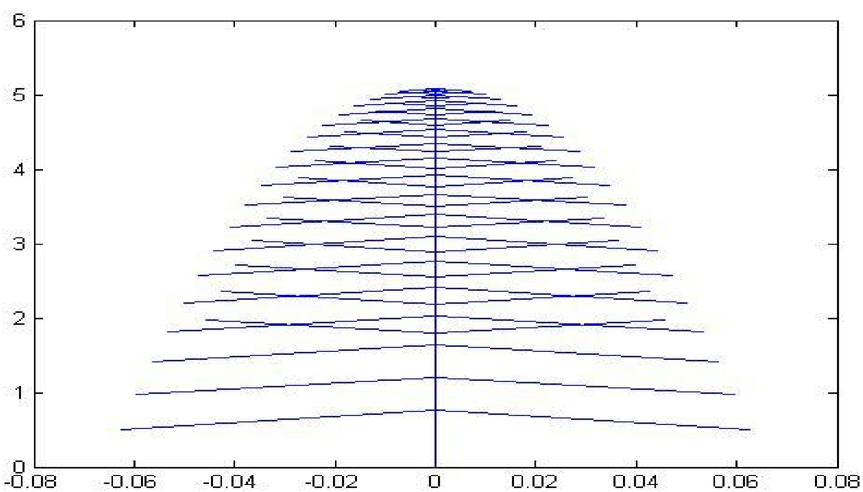
Rhs:

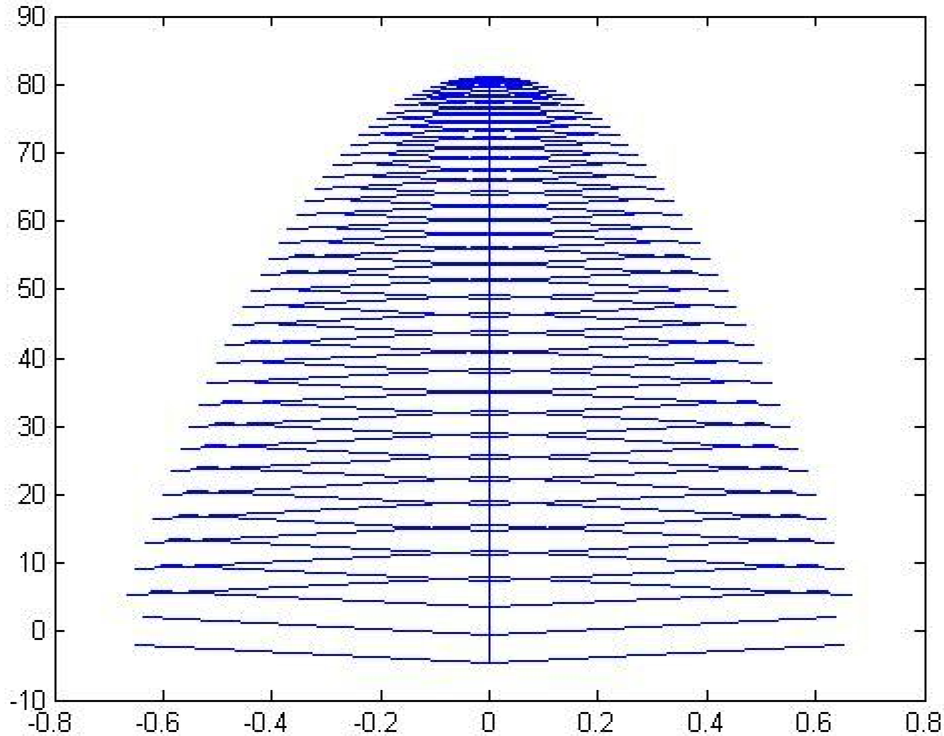
```
function F=rhs(t,u)

G=6.754e-11;
RE=6.4e6;
ME=6e24;
F=zeros(length(u),1);
F(1)=u(2);
F(2)=-G*ME/((u(1)+RE)*(u(1)+RE));
```

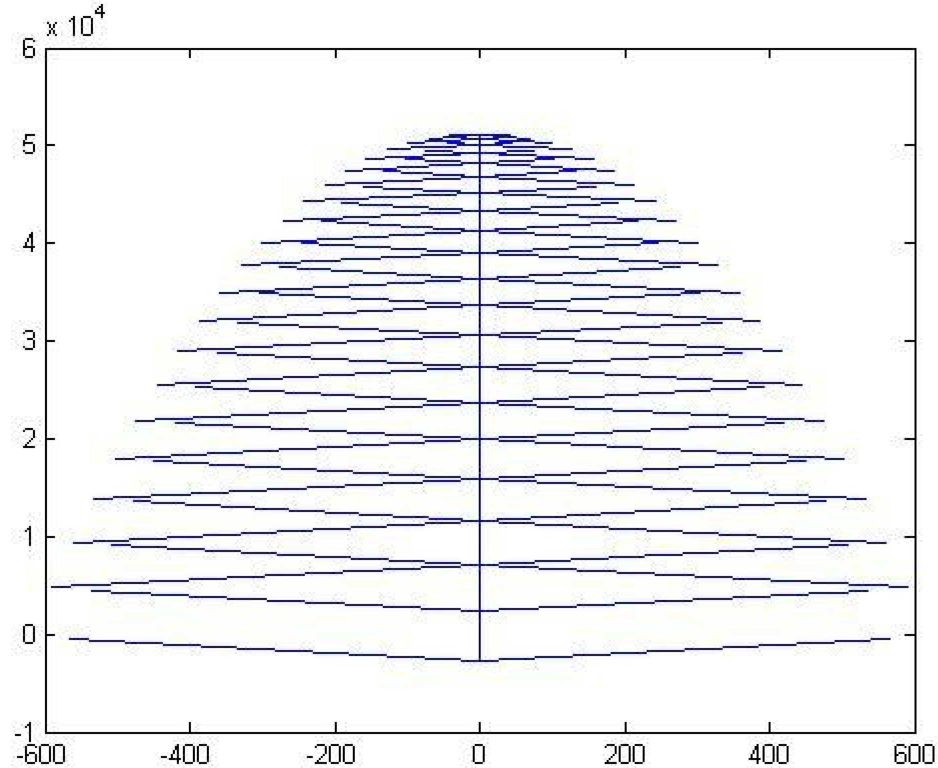
Graph:

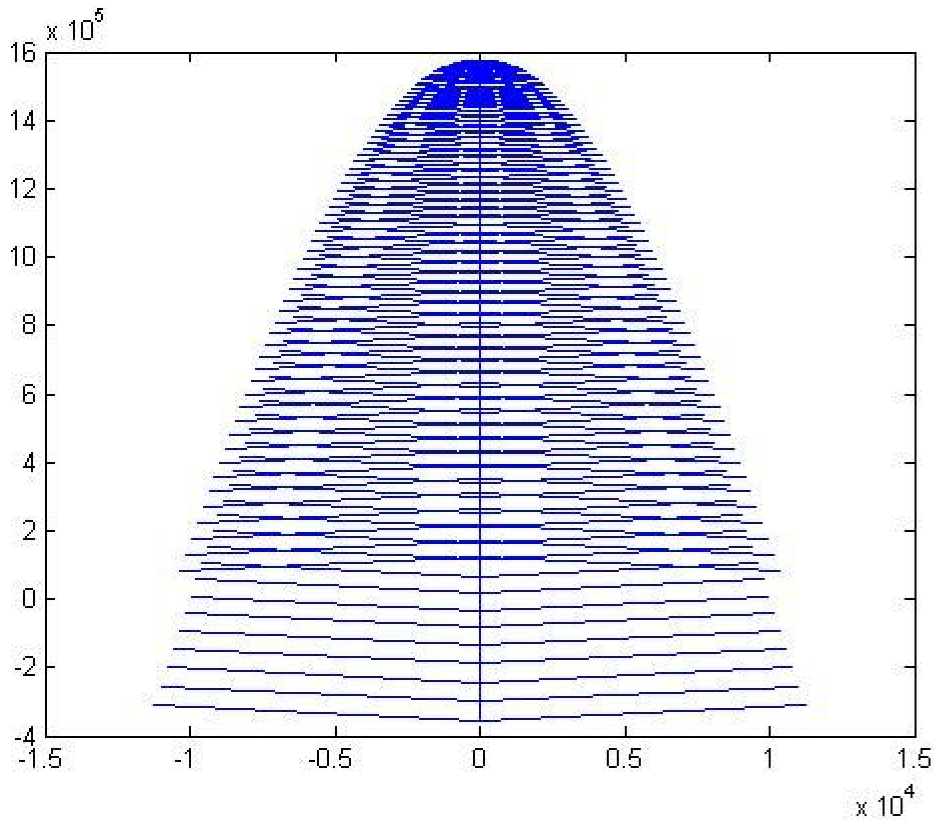
The graph given above shows the velocity vectors at given points, It starts from the initial point (0,0) and then goes up. In the mean-time it's velocity is decreasing. Finally it achieves its maximum possible height. And then it starts falling back. In this time interval the direction of the velocity vector is in opposite direction of the former. Then eventually it will come down.

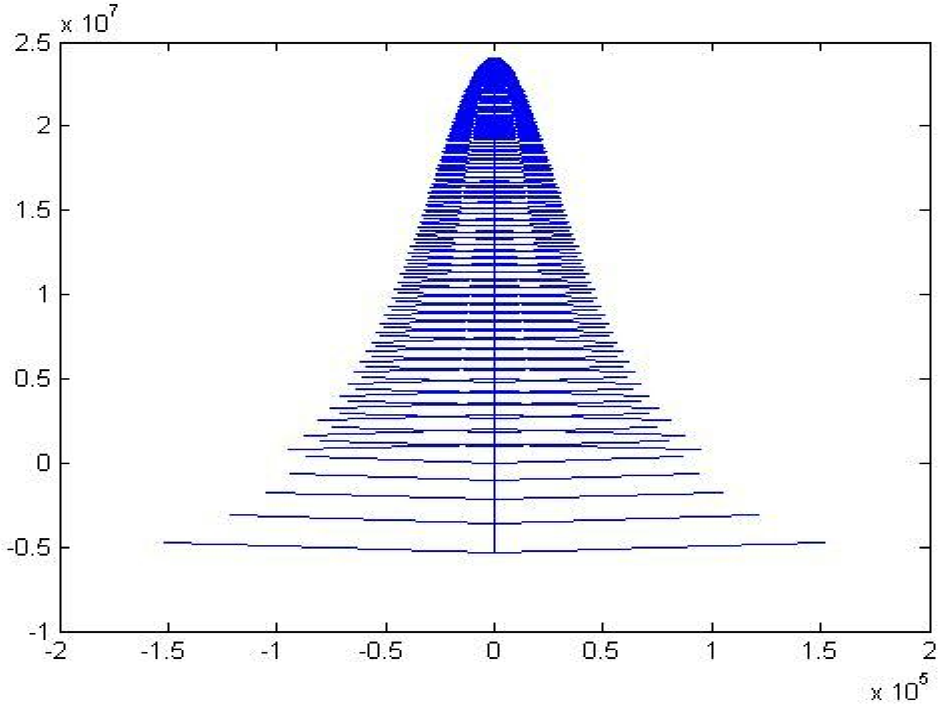

0.06
0.05
0.04
0.03
0.02
0.01
0
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
x 10^{-3}


6
5
4
3
2
1
0
-0.08
-0.06
-0.04
-0.02
0
0.02
0.04
0.06
0.08


90
80
70
60
50
40
30
20
10
0
-10
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8


x 10^4
6
5
4
3
2
1
0
-1
-600
-400
-200
0
200
400
600


x 10^5
16
14
12
10
8
6
4
2
0
-2
-4
-1.5
-1
-0.5
0
0.5
1
1.5
x 10^4



The above graphs are respectively graphs for velocities 1, 10, 40, 1000, 5000 and 10000 m/s. Here the same thing happens that when we throw any ball upward with some initial velocity, it will go upwards and as it moves upwards, its velocity decreases and the arrow show the direction of velocity. At the maximum height, velocity becomes zero. Maximum height depends on the initial velocity, if initial velocity is more, maximum height will also be more.And then it again comes back on ground. In this motion direction of velocity changes, so arrow is opposite. The no. of arrows in graph just depends upon the time period and dt taken by us.

Optional:
You have to run this code for seeing the animation. Given image is last point where the timescale finished.

Code:

```
clear;close all;

timescale=1000;
dt=2;

tstart=0;
tfinal=timescale;
```

```
niter=(tfinal-tstart)/dt;
u0=zeros(2,1);
u0(1)=0; % initial position; %theta(1)=.2;
u0(2)=1000; % initial velocity

G=6.754e-11;
RE=6.4e6;
ME=6e24;
x=zeros(niter,1);
v=zeros(niter,1);
v(1)=u0(2);
x(1)=u0(1);
flag=0;
flag2=0;
for step=1 :niter-1
            v(step+1)=v(step)-((G*ME)/((x(step)+RE)*(x(step)+RE)))*dt;
            x(step+1)=x(step)+v(step+1)*dt;
            if x(step+1)<=0
        if flag==0
       flag=1;

        flag2=flag2+1;
       v(step+1)=((-0.75)*v(step+1));
        end
    end
    if x(step+1)>=0
        flag=0;
        flag2=0;
    end
    if (x(step+1))<-1000 && x(step)<-1000
        break;
    end
    xy=zeros(2,1);
    xy(1)=0;
    xy(2)=x(step+1);
    vxy=zeros(2,1);
    vxy(1)=0;
    vxy(2)=v(step+1);
    plot(vxy,xy,'.')
            axis([-200000 200000 -100000 100000])
            pause(.001)
end
```

Here we cannot download it without running whole graph, hence we just added the last point of the graph. This dot denotes the last position of that point.