Name: Shubham Shivajirao Ware

M.No:221211

Q1.Missing mass fractions

$$b_{i,j} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0.4 & 0 & 0 & 0 \\ 0.3 & 0.4 & 0 & 0 \\ 0.2 & 0.3 & 0.3 & 0 \\ 0.1 & 0.3 & 0.7 & 1 \end{bmatrix}$$

governing differential equation:

$$\frac{dm_i}{dt} = -s_i m_i(t) + \sum_{j=1}^{i-1} s_j m_j(t) b_{ij} \quad \ldots(\text{eq } 1.1)$$

Note: (expanded form is shown below as a part of a code to avoid repetition)

Q2. Mat-lab code:

```
%first function handles are assigned to function fn1 to fn4 by expanding equation 1.1 Using function handle we can save time of defining function in different script and then recalling it in our program, following function handle is called Anonymous function handle.

fn1=@(t,m1)                   -s(1)*m1;

fn2=@(t,m1,m2)                -s(2)*m2 +s(1)*m1*b(2,1);

fn3=@(t,m1,m2,m3)             -s(3)*m3 +s(1)*m1*b(3,1) +s(2)*m2*b(3,2) ;

fn4=@(t,m1,m2,m3,m4)          -s(4)*m4 +s(1)*m1*b(4,1) +s(2)*m2*b(4,2)
                              +s(3)*m3*b(4,3);

fn5=@(t,m1,m2,m3,m4,m5) -s(5)*m5+s(1)*m1*b(5,1) +s(2)*m2*b(5,2)
                        +s(3)*m3*b(5,3) +s(4)*m4*b(5,4);

%the above equations also show the expanded form of governing partial differential equations,

% initial conditions for  mass variables are defined respectively

m1(1)=100;

m2(1)=0;

m3(1)=0;

m4(1)=0;

m5(1)=1;

t(1)=0;


%solving differential equation by RK-4 method, I have generated loop for Δt  time increment
```

```
for i=1:tend/dt
    t(i+1)=t(i)+dt; %updating time interval



%now finding out RK-4 constants for each equation
    k1=fn1(t(i)     , m1(i))                           ;
    l1=fn2(t(i)     , m1(i), m2(i))                    ;
    n1=fn3(t(i)     , m1(i), m2(i), m3(i))             ;
    o1=fn4(t(i)     , m1(i), m2(i), m3(i), m4(i)       ;
    p1=fn5(t(i)     , m1(i), m2(i), m3(i), m4(i), m5(i));

    k2=fn1(t(i)*dt/2, m1(i)+dt/2*k1)                                              ;
    l2=fn2(t(i)*dt/2, m1(i)+dt/2*l1, m2(i)+dt/2*l1)                               ;
    n2=fn3(t(i)*dt/2, m1(i)+dt/2*n1, m2(i)+dt/2*n1, m3(i)+dt/2*n1)                ;
    o2=fn4(t(i)*dt/2, m1(i)+dt/2*o1, m2(i)+dt/2*o1, m3(i)+dt/2*o1, m4(i)+dt/2*o1) ;
p2=fn5(t(i)*dt/2,m1(i)+dt/2*p1,m2(i)+dt/2*p1,m3(i)+dt/2*p1,m4(i)+dt/2*p1,m5(i)+dt/2*p1);

  k3=fn1(t(i)*dt/2, m1(i)+dt/2*k2)                                              ;
  l3=fn2(t(i)*dt/2, m1(i)+dt/2*l2, m2(i)+dt/2*l2)                               ;
  n3=fn3(t(i)*dt/2, m1(i)+dt/2*n2, m2(i)+dt/2*n2, m3(i)+dt/2*n2)                ;
  o3=fn4(t(i)*dt/2, m1(i)+dt/2*o2, m2(i)+dt/2*o2, m3(i)+dt/2*o2, m4(i)*dt/2*o2) ;
p3=fn5(t(i)*dt/2,m1(i)+dt/2*p2,m2(i)+dt/2*p2,m3(i)+dt/2*p2,m4(i)*dt/2*p2,m5(i)+dt/2*p2);

    k4=fn1(t(i)*dt  , m1(i)+dt*k3)                                       ;
    l4=fn2(t(i)*dt  , m1(i)+dt*l3, m2(i)+dt*l3)                          ;
    n4=fn3(t(i)*dt  , m1(i)+dt*n3, m2(i)+dt*n3, m3(i)+dt*n3)             ;
    o4=fn4(t(i)*dt  , m1(i)+dt*o3, m2(i)+dt*o3, m3(i)+dt*o3, m4(i)+dt*o3)  ;
    p4=fn5(t(i)*dt  , m1(i)+dt*p3, m2(i)+dt*p3, m3(i)+dt*p3, m4(i)+dt*p3, m5(i)+dt*p3);
```

```
%obtaining next value of mass using Runge-Kutta (RK-4) method
    m1(i+1)=m1(i)+dt/6*(k1+2*k2+2*k3+k4);
    m2(i+1)=m2(i)+dt/6*(l1+2*l2+2*l3+l4);
    m3(i+1)=m3(i)+dt/6*(n1+2*n2+2*n3+n4);
    m4(i+1)=m4(i)+dt/6*(o1+2*o2+2*o3+o4);
    m5(i+1)=m5(i)+dt/6*(p1+2*p2+2*p3+p4);
end
```

Q.4

RESULT

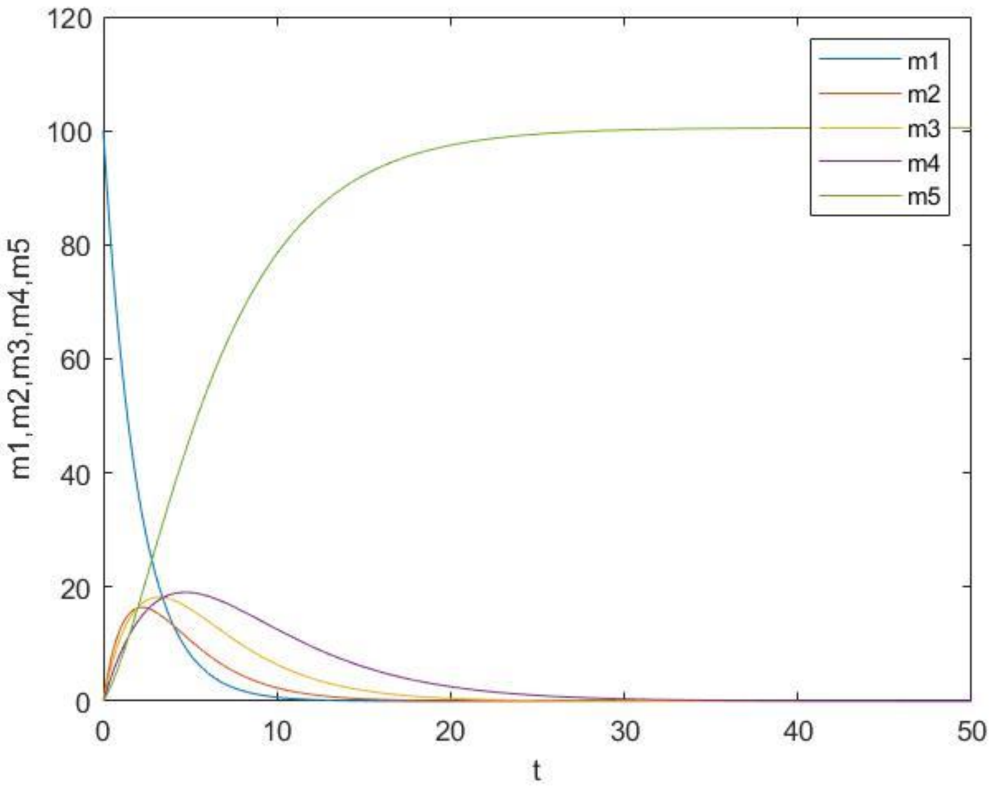

Fig 1.2

Q.4.

Time requited for 99.9% of sand to come in mass fraction m5; according to my calculation was around **22 time intervals.** Which is clearly seen in fig 1.2

Q.5

As Δt decreases fineness of plot increases, so is the time for calculation. However the optimal time interval would be one comparable to revolutions of mill used for grinding the particles, hence if mill is rotating 100 times per second optimal Δt would be 1/100 sec.