**Experiment - 02**

**\*\*Write a sci lab code for the following problems: \*\***

**1.Plotting of a Scalar Field f (x, y) = x^2y.**

**Code:-** clc

clear

clf

function **z**=scalarfield(**x**, **y**)

**z**=(**x**.^2).\***y**

endfunction

x=linspace(-10,10,100)

y=linspace(-10,10,100)

[X,Y]=meshgrid(x,y)

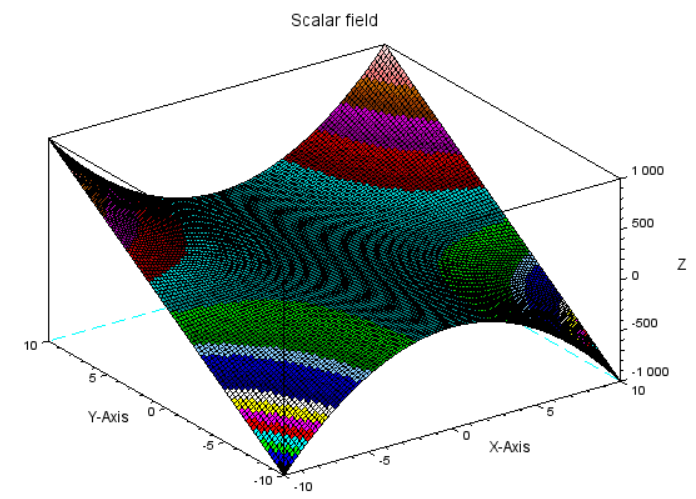
z=scalarfield(X, Y)

surf(x, y, z)

xtitle('Scalar field', 'X-Axis', 'Y-Axis')

Colorbar

**Output**:-

****

**2.** **Plotting of a Scalar Field f(x, y) = xe^-(x^2+y^2).**

**Code:-** clc

clear

clf

function **z**=scalarfield(**x**, **y**)

**z**=x.\*exp(-x.^2-y.^2)

endfunction

x=linspace(6,-6100)

y=linspace(6,6,100)

[X,Y]=meshgrid(x,y)

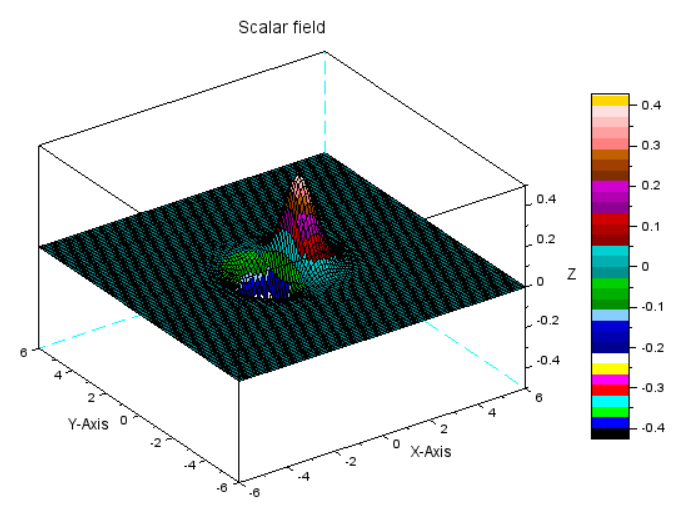
z=scalarfield(X, Y)

surf(x, y, z)

xtitle('Scalar field', 'X-Axis', 'Y-Axis')

colorbar

**Output**:-

****

**3. Plotting of a Vector Field f( x, y) = (x, y)**

**Code:-** clc

clear

clf

function [**Zx**, **Zy**]=vfield(**x**, **y**)

**Zx**=**x**

**Zy**=**y**

endfunction

x=-4:.1:4

y=-4:.1:4

[X,Y]=ndgrid(x,y)

[Zx,Zy]=vfield(X,Y)

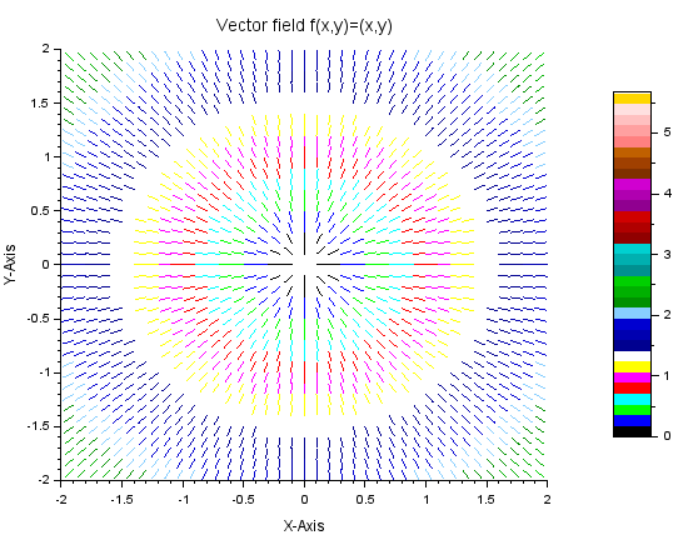
champ(x,y,Zx,Zy,0.2,rect=[-2,-2,2,2])

gce().colored="on"

xtitle('Vector field f(x,y)=(x,y)','X-Axis','Y-Axis');

colorbar

**Output**:-

****

**4. Plotting of a Vector Field f( x, y) = (-y, x)**

**Code:-** clc

clear

clf

function [**Zx**, **Zy**]=vfield(**x**, **y**)

**Zx**=-y

**Zy**=x

endfunction

x=-4:.1:4

y=-4:.1:4

[X,Y]=ndgrid(x,y)

[Zx,Zy]=vfield(X,Y)

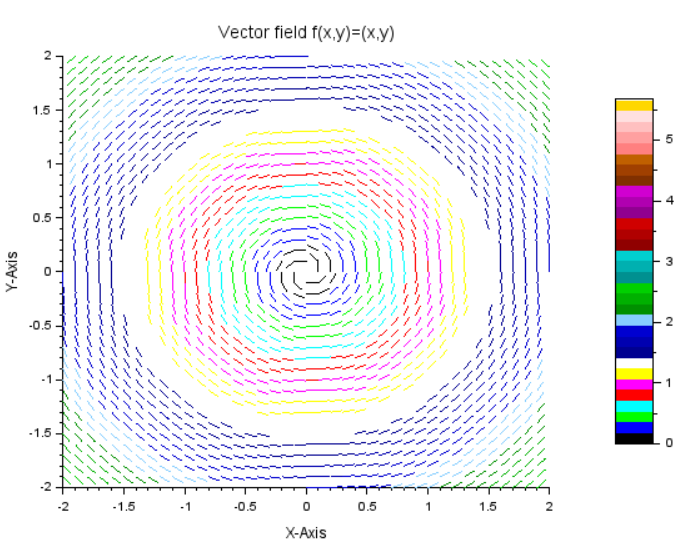
champ(x,y,Zx,Zy,0.2,rect=[-2,-2,2,2])

gce().colored="on"

xtitle('Vector field f(x,y)=(-y,x)','X-Axis','Y-Axis')

colorbar

**Output**:-

****

**5. Plotting of a Vector Field f( x, y) = (sin y, cos x).**

**Code:-** clc

clear

clf

function [**Zx**, **Zy**]=vfield(**x**, **y**)

**Zx**=sin (**y**)

**Zy**=cos (**x**)

endfunction

x=linspace(-1,1,100)\*2\*%pi

y=linspace(-1,1,100)\*2\*%pi

[X,Y]=ndgrid(x,y)

[Zx,Zy]=vfield(X,Y)

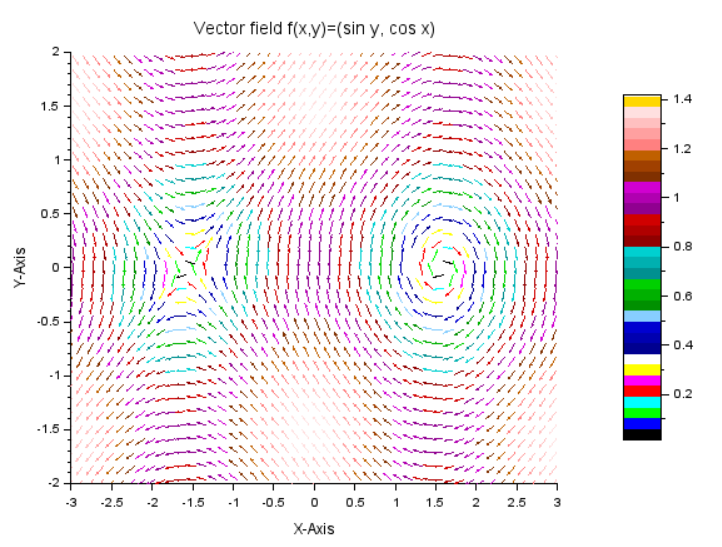
champ(x,y,Zx,Zy,0.5,rect=[-3,-2,3,2])

gce().colored="on"

xtitle('Vector field f(x,y)=(sin y, cos x)','X-Axis','Y-Axis')

colorbar

**Output**:-

****

**6.** **The gradient of a scalar Field f( x, y) = x^2\*y.**

**Code:-**

clc

clear

clf

function [**z**, **DZx**, **DZy**]=scalarfield(**x**, **y**)

**z**=**x**.^2.\***y**

**DZx**=2\***x**.\***y**

**DZy**=**x**.^2

endfunction

x=linspace(-5,5,100)

y=linspace(-5,5,100)

[X,Y]=meshgrid(x,y)

[z,DZx,DZy]=scalarfield(X,Y)

surf(x,y,z)

xtitle('scalar field f(x,y)=x^2y','X-Axis','Y-Axis')

scf

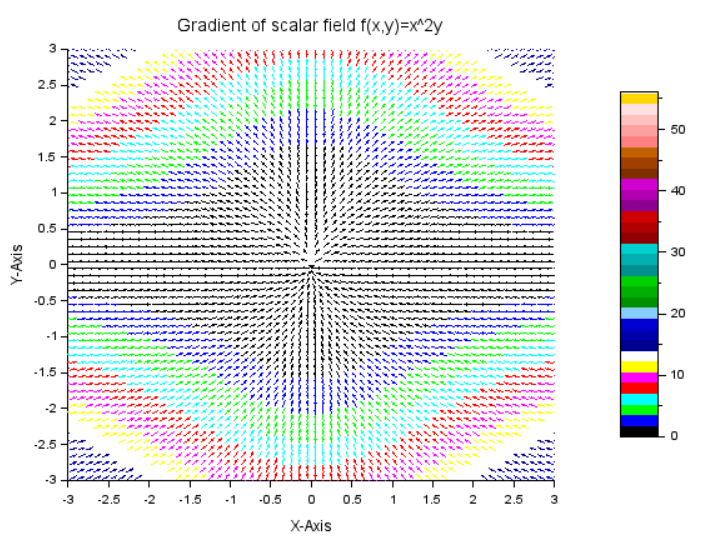
champ(x,y,DZx,DZy,0.5,rect=[-3,-3,3,3])

gce().colored="on"

xtitle('Gradient of scalar field f(x,y)=x^2y','X-Axis','Y-Axis');

colorbar

**Output**:-

****

**7. The gradient of a Scalar Field f(x, y) = xe^-(x^2+y^2).**

**Code:-**

clc

clear

clf

function [**z**, **DZx**, **DZy**]=scalarfield(**x**, **y**)

**z**=**x**.\*exp(-x.^2-y^2)

**DZx**=exp(-x.^2-y.^2)-2\*x.^2.\*exp(-x.^2-y.^2)

**DZy**=-2\***x**.\*y.\*exp(-x.^2-y.^2)

endfunction

x=linspace(-4,4,100)

y=linspace(-4,4,100)

[X,Y]=meshgrid(x,y)

[z,DZx,DZy]=scalarfield(X,Y)

surf(x,y,z)

xtitle('scalar field ','X-Axis','Y-Axis')

colorbar

scf

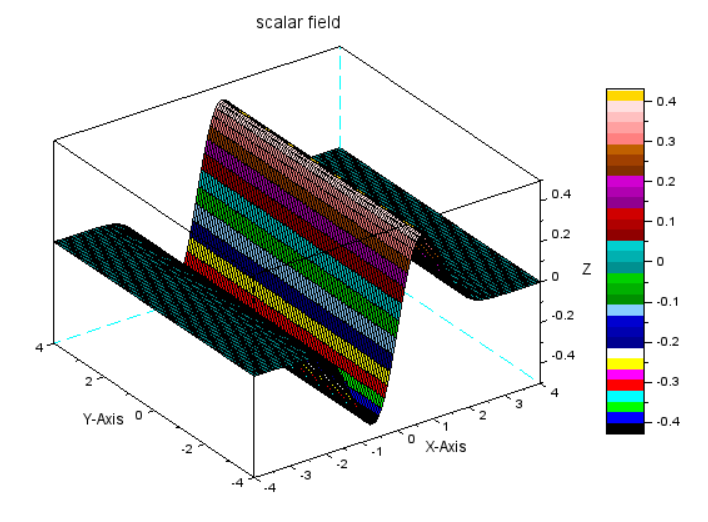
champ(x,y,DZx,DZy,0.3,rect=[-2,-2,2,2])

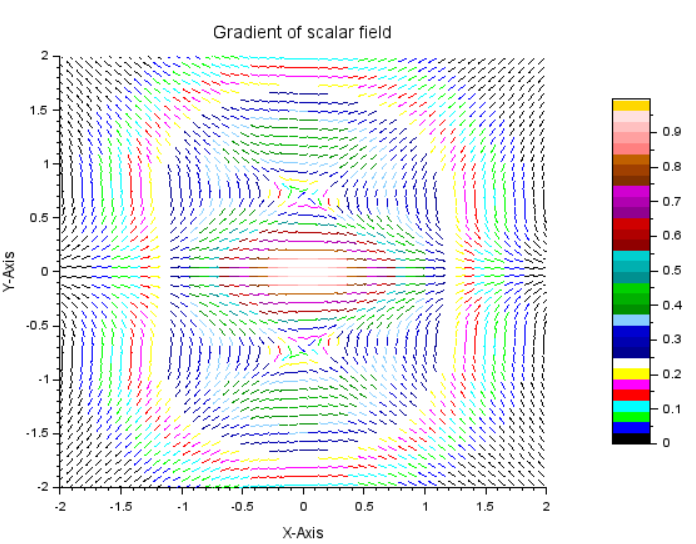
gce().colored="on"

xtitle('Gradient of scalar field f(x,y)=x^2y','X-Axis','Y-Axis')

colorbar

**Output**:-

****

****

**8.Divergence of a Vector Field f( x, y) = (-y, x)**

**Code:-**

clc

clear

clf

function [**Zx**, **Zy, Div**]=vfield(**x**, **y**)

**Zx**=-y

**Zy**=x

Div=0\*x

endfunction

x=-4:.1:4

y=-4:.1:4

[X,Y]=meshgrid(x,y)

[Zx,Zy,Div]=vfield(X,Y)

surf(x,y,Div)

xtitle('Divergence of Vector field f(x,y)=(y,x)','X-Axis','Y-axis')

colorbar

scf

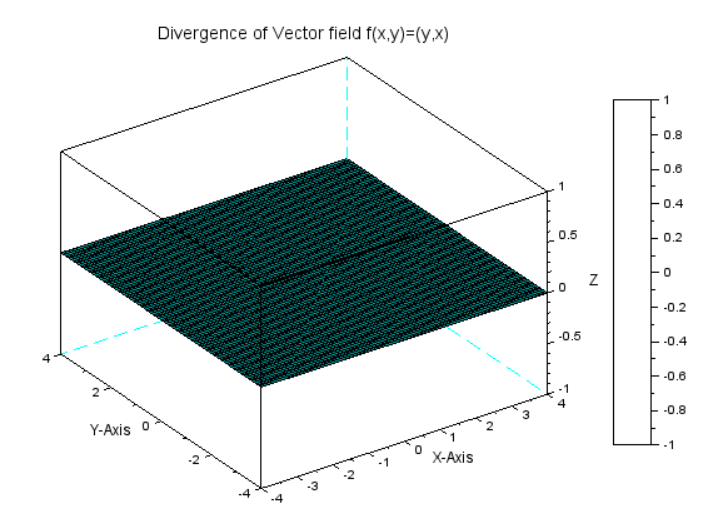
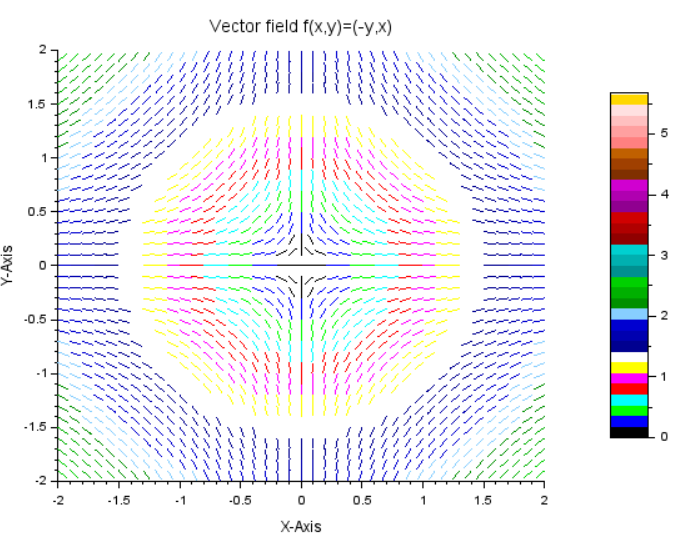
champ(x,y,Zx,Zy,0.2,rect=[-2,-2,2,2])

gce().colored="on"

xtitle('Vector field f(x,y)=(-y,x)','X-Axis','Y-Axis')

colorbar

**Output**:-

****

**9. Divergence of a Vector Field f( x, y) = (sin y, cos x).**

**Code:-** **:-**

clc

clear

clf

function [**Zx**, **Zy, Div**]=vfield(**x**, **y**)

**Zx**=sin(x)

**Zy**=cos(y)

Div=cos(x)-sin(y)

endfunction

x=-1:1:50\*%pi

y=-1:1:50\*%pi

[X,Y]=ndgrid(x,y)

[Zx,Zy,Div]=vfield(X,Y)

surf(x,y,Div)

xtitle('Divergence of Vector field','X-Axis','Y-axis')

colorbar

scf

champ(x,y,Zx,Zy,0.2,rect=[-3,-%pi,2.5,%pi])

gce().colored="on"

xtitle('Vector field','X-Axis','Y-Axis')

colorbar

**Experiment - 03**

**\*\*Write a sci lab code for the following problems: \*\***

**Write a script file to determine the Linear independence (LI) of vectors and plot these vectors in their respective space.**

**1.(i) (0,1), (1,0)**

**Code:-**

clc

clear

clf

a=[0,1]

b=[1,0]

A=[a;b]

n=input("Enter the numbers of vectors:")

r=rank(A)

if r==n then

disp("Linear Independent.")

else

disp("Linear dependent.")

end

plot2d([0,0],[0,0],[0,0])

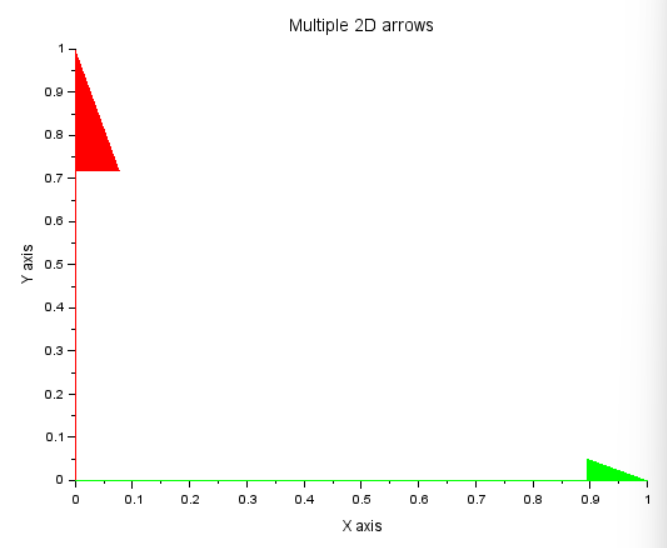
xarrows([0,0],[0,1],4,5)

xarrows([0,1],[0,0],2,3)

xtitle( 'Multiple 2D arrows', 'X axis', 'Y axis')

**Output:-** Enter the numbers of vectors:2

"Linear Independent."



**(ii) (2,4), (1,2)**

**Code:-**

clc

clear

clf

a=[2,4]

b=[1,2]

A=[a;b]

n=input("Enter the numbers of vectors:")

r=rank(A)

if r==n then

disp("Linear Independent.")

else

disp("Linear dependent.")

end

plot2d([0,0],[0,0],[0,0])

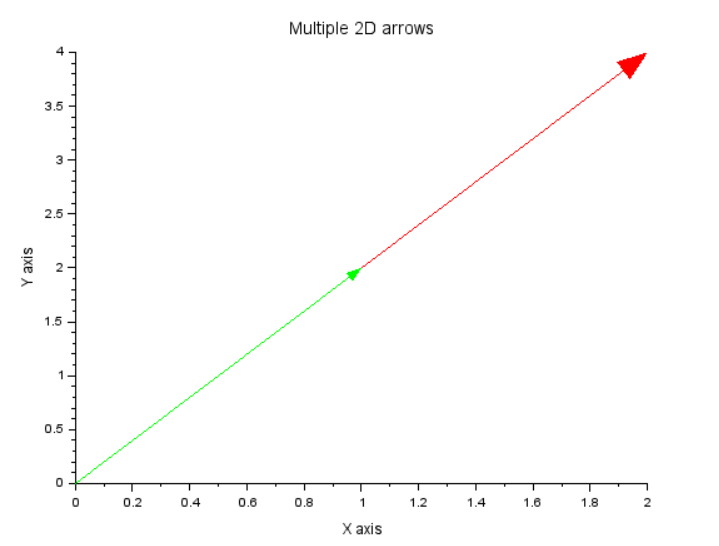
xarrows([0,2],[0,4],4,5)

xarrows([0,1],[0,2],2,3)

xtitle( 'Multiple 2D arrows', 'X axis', 'Y axis')

**Output:-** Enter the numbers of vectors:2

"Linear dependent."

****

**(iii) (1,1), (1,3), (2,4)**

**Code:-**

clc

clear

clf

a=[1,1]

b=[1,3]

c=[2,4]

A=[a;b;c]

n=input("Enter the numbers of vectors:")

r=rank(A)

if r==n then

disp("Linear Independent.")

else

disp("Linear dependent.")

end

plot2d([0,0],[0,0],[0,0])

xarrows([0,1],[0,1],4,5)

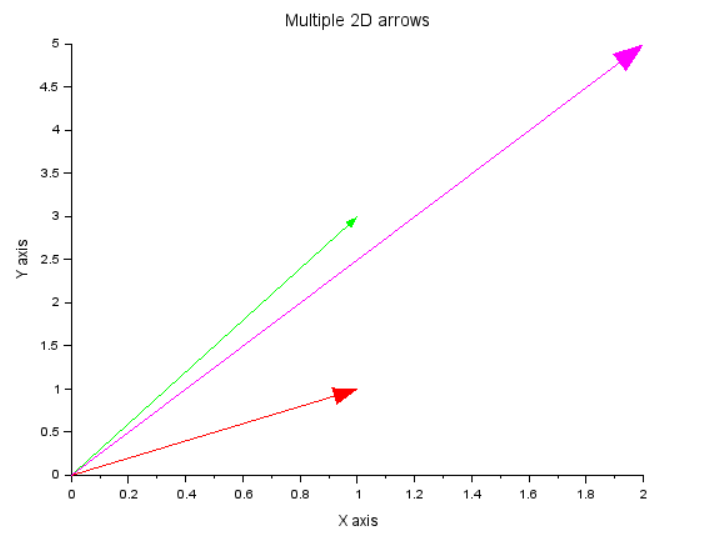
xarrows([0,1],[0,3],2,3)

xarrows([0,2],[0,4],2,3)

xtitle( 'Multiple 2D arrows', 'X axis', 'Y axis')

**Output:-** Enter the numbers of vectors:2

"Linear dependent."



**(iv) (1,2,3), (1,2,4)**

**Code:-**

clc

clear

clf

a=[1,2,3]

b=[1,2,4]

A=[a;b]

n=input("Enter the numbers of vectors:")

r=rank(A)

if r==n then

disp("Linear Independent.")

else

disp("Linear dependent.")

end

plot3d([0,0],[0,0],[0,0],[0,0])

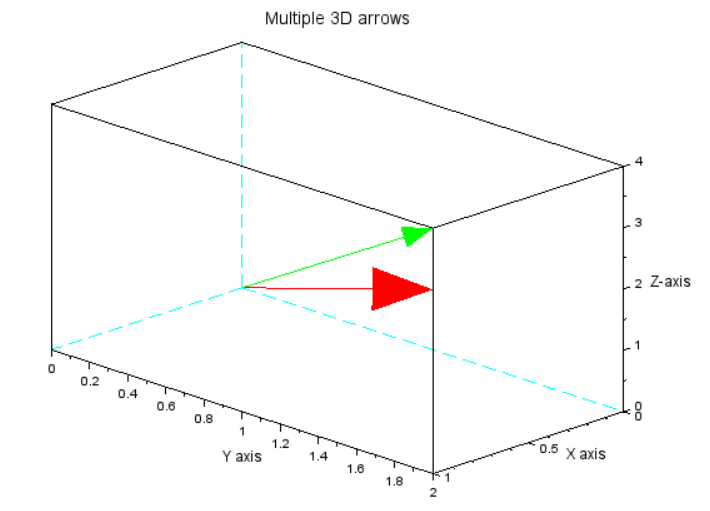
xarrows([0,1],[0,2],[0,3],4,5)

xarrows([0,1],[0,2],[0,4],,2,3)

xtitle( 'Multiple 3D arrows', 'X axis', 'Y axis')

**Output:-** Enter the numbers of vectors:2

"Linear Independent."



**(v) (1,1,0), (1,0,1), (0,1,1)**

**Code:-**

clc

clear

clf

a=[1,1,0]

b=[1,0,1]

c=[0,1,1]

A=[a;b;c]

n=input("Enter the numbers of vectors:")

r=rank(A)

if r==n then

disp("Linear Independent.")

else

disp("Linear dependent.")

end

plot3d([0,0],[0,0],[0,0],[0,0])

xarrows([0,1],[0,1],[0,0],4,5)

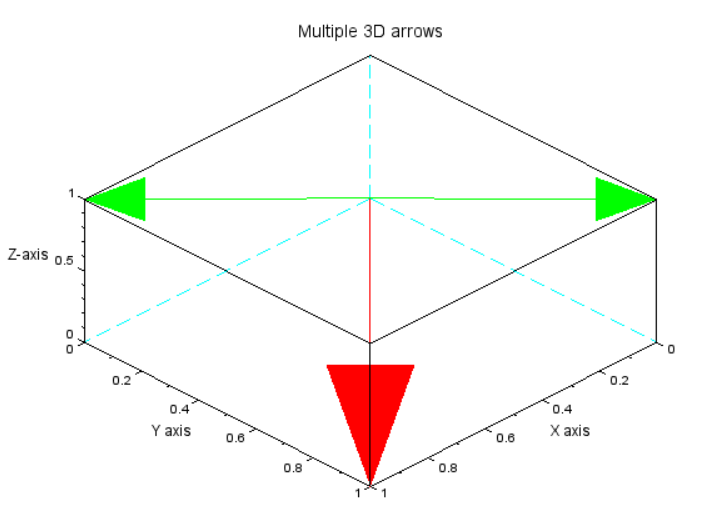
xarrows([0,1],[0,0],[0,1],,2,3)

xarrows([0,0],[0,1],[0,1],,2,3)

xtitle( 'Multiple 3D arrows', 'X axis', 'Y axis')

**Output:-** Enter the numbers of vectors:3

"Linear Independent."



**(vi) (2,2,1), (1,-1,1), (1,0,1)**

**Code:-**

clc

clear

clf

a=[2,2,1]

b=[1,-1,1]

c=[1,0,1]

A=[a;b;c]

n=input("Enter the numbers of vectors:")

r=rank(A)

if r==n then

disp("Linear Independent.")

else

disp("Linear dependent.")

end

plot3d([0,0],[0,0],[0,0],[0,0])

xarrows([0,2],[0,2],[0,1],4,5)

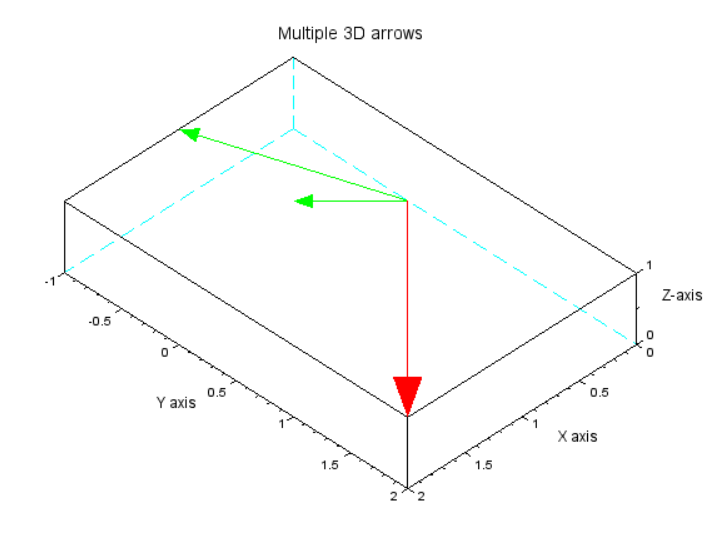
xarrows([0,1],[0,-1],[0,1],,2,3)

xarrows([0,1],[0,0],[0,1],,2,3)

xtitle( 'Multiple 3D arrows', 'X axis', 'Y axis')

**Output:-** Enter the numbers of vectors:3

"Linear Independent."

****

**(vii)(1,2,3,1), (2,1,-1,1), (4,5,5,3), (5,4,1,3)**

**Code:-**

clc

clear

clf

a=[1.2,3,1]

b=[2,1,-1,1]

c=[4,5,5,3]

d=[5,4,1,3]

A=[a;b;c;d]

n=input("Enter the numbers of vectors:")

r=rank(A)

if r==n then

disp("Linear Independent.")

else

disp("Linear dependent.")

end

**Output:-** Enter the numbers of vectors:4

"Linear dependent."

**(viii) (1,2,3), (1,2,4), (0, 0, 0)**

**Code:-** clc

clear

clf

a=[1.2,3]

b=[1,2,4]

c=[0,0,0]

A=[a;b;c]

n=input("Enter the numbers of vectors:")

r=rank(A)

if r==n then

disp("Linear Independent.")

else

disp("Linear dependent.")

end

plot3d([0,0],[0,0],[0,0],[0,0])

xarrows([0,1],[0,2],[0,3],4,5)

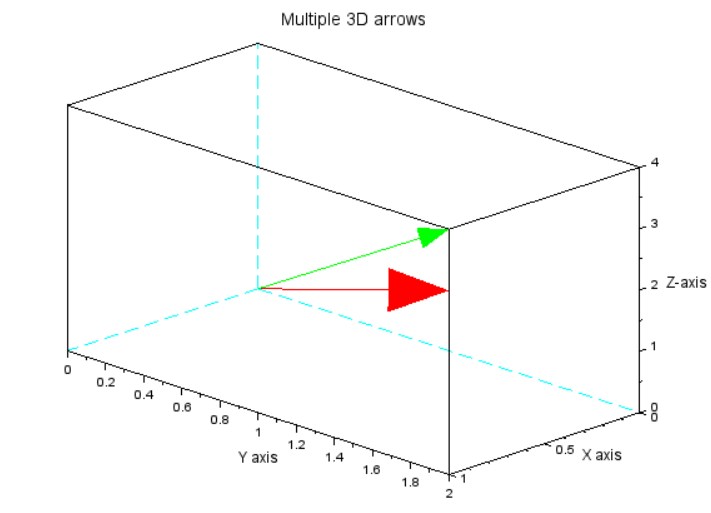
xarrows([0,1],[0,2],[0,4],,2,3)

xarrows([0,0],[0,0],[0,0],,2,3)

xtitle( 'Multiple 3D arrows', 'X axis', 'Y axis')

**Output:-** Enter the numbers of vectors:3

"Linear dependent."

****