Experiment :-04

Experiment No. 4 Representation of the Gradient of a scalar field, Divergence and Curl of Vector fields.

1. Write a matlab program to find the gradient of the following scalar fields and verify analytically also

   (a) $T = x \exp(-x^2 - y^2)$ at $(x, y) \equiv (1, 2)$

   (b) $V = e^{-z} \sin 2x \cosh y$

   (c) $U = \rho^2 z \cos 2\phi$

   (d) $W = 10 r \sin^2\theta \cos\phi$

   (e) $T1 = x^2 y + xyz$

   (f) $V1 = \rho z \sin\phi + z^2 \cos^2\phi + \rho^2$

   (g) $W1 = \cos\theta \sin\phi \ln r + r^2 \phi$

2. Given $W = x^2 y^2 + xyz$, Compute $\vec{\nabla} W$ and the directional derivative $\frac{dW}{dl}$ in the direction $3\hat{a}_x + 4\hat{a}_y + 12\hat{a}_z$ at $(2, -1, 0)$.
   Write Matlab Program for finding $\vec{\nabla} W$ and directional directive.

3. Determine the divergence of these vector fields:

   (a) $\vec{P} = x^2 yz\, \hat{a}_x + xz\, \hat{a}_z$

   (b) $\vec{R} = x \hat{a}_x + 2y^2 \hat{a}_y + 3z^3 \hat{a}_z$

   (c) $\vec{Q} = \rho \sin\phi\, \hat{a}_\rho + \rho^2 z\, \hat{a}_\phi + z \cos\phi\, \hat{a}_z$

   (d) $\vec{T} = \frac{1}{r^2} \cos\theta\, \hat{a}_r + r \sin\theta \cos\phi\, \hat{a}_z + \cos\theta\, \hat{a}_\phi$

   Write the Matlab program for the same.

4. Write the Matlab program determining the divergence of vector field $\vec{A} = yz\, \hat{a}_x + 4xy\, \hat{a}_y + y\hat{a}_z$ at $(1, -2, 3)$

5. Determine the curl of each of the vector field $\vec{P}, \vec{Q}, \vec{T}$ in Ques. 3. and also write the matlab Programs

6. Find the Curl of the gradient of a given scalar field f given as $f = x^2 + y^2 + z^2$, that is $\nabla \times \nabla f$. Write Matlab Program.

7. Find the divergence of the curl of a given vector field $\vec{P}$ given in Ques 3, that is $(\vec{\nabla} \cdot \vec{\nabla} \times \vec{A})$ and write also the MatLab Program

8. Evaluate ~~and plot the~~ the curl of Vector field $y^2 \hat{a}_x + x^2 \hat{a}_y$. Using Quiver Command, plot the $y^2 \hat{a}_x + x^2 \hat{a}_y$. Using Quiver Command, plot the X and Y given by vector field; evaluating the Curl over for
   $[X, Y] = \text{Meshgrid}(-1:0.1:1, -1:0.1:1)$

   (a) $y^2 \hat{a}_x + x^2 \hat{a}_y$ (b) $\sin(\pi y)\, \hat{a}_x + \sin(\pi x)\, \hat{a}_y$

(9) Find the numerical gradient using Contour and Quiver for $z = x \exp(-x^2 - y^2)$, and observe the gradient and velocity curve at the output.

1.
```
syms x y
f=x*exp(-x*x-y*y);
t=gradient(f,[x,y])
%%
syms x y z
f= exp(-z)*sin(2*x)*cosh(y);
t=gradient(f,[x,y,z])
%%
syms rho phi z
f=(rho^2)*(z)*cos(2*phi)
t=gradient(f,[rho,phi,z])
%%
syms r theta phi
f=10*r*((sin(theta))^2)*cos(phi);
t=gradient(f,[r,theta,phi])
%%
syms x y z
f=x^2*y+x*y*z;
t=gradient(f,[x,y,z])
%%
syms rho phi z
f=rho*z*sin(phi)+z^2*(cos(phi)^2)+rho^2 ;
t=gradient(f,[rho,phi,z])
%%
syms r theta phi
f=cos(theta)*sin(phi)*log(r)+r^2*(phi);
t=gradient(f,[r,theta,phi])
```

Output:-

```
>> gradi

t = exp(- x^2 - y^2) - 2*x^2*exp(- x^2 - y^2)

                -2*x*y*exp(- x^2 - y^2)

 t = 2*cos(2*x)*exp(-z)*cosh(y)

   sin(2*x)*exp(-z)*sinh(y)

  -sin(2*x)*exp(-z)*cosh(y)

 f =rho^2*z*cos(2*phi)

 t =

 2*rho*z*cos(2*phi)

 -2*rho^2*z*sin(2*phi)

     rho^2*cos(2*phi)
```

t = 10*cos(phi)*sin(theta)^2

20*r*cos(phi)*cos(theta)*sin(theta)

-10*r*sin(phi)*sin(theta)^2

t = 2*x*y + y*z

x^2 + z*x

x*y

t =

2*rho + z*sin(phi)

- 2*cos(phi)*sin(phi)*z^2 + rho*cos(phi)*z

2*z*cos(phi)^2 + rho*sin(phi)

t =

2*phi*r + (cos(theta)*sin(phi))/r

-log(r)*sin(phi)*sin(theta)

r^2 + cos(phi)*cos(theta)*log(r)

Qno.2

```
syms x y z
W(x,y,z)= (x^2)*(y^2) + x*y*z;
b=gradient(W,[x,y,z])
b(2,-1,0)
a=[3,4,12]
c=(1/13)*dot(b(2,-1,0),a)
```

```
Output:-
b(x, y, z) =

 2*x*y^2 + z*y
 2*y*x^2 + z*x
             x*y


ans =

  4
 -8
```

-2

a =

     3   4   12

c =

-44/13

Qno.3

```
% a) P=((x^2)*y*z) ax + 0 ay + x*z az
syms x y z
div=divergence([(x^2)*y*z 0 x*z], [x,y,z])
%%
% b) R=[x 2*y*y 3*(z^3) ]
syms x y z
div=divergence([x 2*y*y 3*(z^3)],[x,y,z])
%%
% c) Q=[rho*sin(phi) (rho^2)*z z*cos(phi)]
syms rho phi z
div=divergence_sym([rho*sin(phi) (rho*rho)*z
z*cos(phi)],[rho,phi,z],'Cylindrical')
%
% d) T=[(r^-2)*cos(theta) r*sin(theta)*cos(phi) cos(theta)]
syms r theta phi
div=divergence_sym([(1/r*r)*cos(theta) r*sin(theta)*cos(phi)
cos(theta)],[r,theta,phi],'Spherical')
```

Output:-
exp4c

div =

x + 2*x*y*z

div =

9*z^2 + 4*y + 1

div =

cos(phi) + 2*sin(phi)

div =

(2*cos(theta))/r + 2*cos(phi)*cos(theta)

Qno.04

```
% A = [y*z 4*x*y y] at point (1,-2,3)
syms x y z
div(x,y,z)=divergence([y*z 4*x*y y],[x,y,z])
div(1,-2,3)
```

Output:-

```
div(x, y, z) =

4*x


ans =

4
```

Qno.05

```
% determine the curl of vector
% a) P=((x^2)*y*z) ax + 0 ay + x*z az
syms x y z
q(x,y,z)=curl([(x^2)*y*z 0 x*z],[x,y,z])
a=curl_sym([(x^2)*y*z 0 x*z],[x,y,z],'Cartesian')
%%
% b) Q=[rho*sin(phi) (rho^2)*z z*cos(phi)]
syms rho phi z
b=curl_sym([rho*sin(phi) (rho^2)*z z*cos(phi)],[rho,phi,z],'Cylindrical')
%%
% T=[(r^-2)*cos(th) r*sin(th)*cos(phi) cos(th)]
syms r th phi
a=curl_sym([(r^-2)*cos(th) r*sin(th)*cos(phi)
cos(th)],[r,th,phi],'Spherical')
```

Output:-

```
q(x, y, z) =

         0
 y*x^2 - z
    -x^2*z


a =

         0
 y*x^2 - z
    -x^2*z
```

```
b =

[ - rho^2 - (z*sin(phi))/rho, 0, 3*rho*z - cos(phi)]


a =

[ (cos(th)^2 - sin(th)^2 + r*sin(phi)*sin(th))/(r*sin(th)), -
cos(th)/r, (sin(th)/r^2 + 2*r*cos(phi)*sin(th))/r]
```

Qno.06

```
%f= x^2 + y^2 + z^2
syms x y z
f= x^2 + y^2 + z^2
g(x,y,z)= gradient_sym(f,[x,y,z],'Cartesian')
c= curl_sym(g,[x,y,z],'Cartesian')
```

Output:-

```
f =

x^2 + y^2 + z^2


g(x, y, z) =

 2*x
 2*y
 2*z


c(x, y, z) =

 0
 0
 0
```

Qno.07

```
% P= [(x^2)*y*z 0 x*z] to find the divergence of a curl of P
syms x y z
c=curl_sym([(x^2)*y*z 0 x*z],[x,y,z],'Cartesian')
d=divergence_sym(c,[x,y,z],'Cartesian')
```

Output:- c =

```
          0
 y*x^2 - z
    -x^2*z
```

```
d =

0
```

Qno.08

```
% To find the curl of a vector field [y^2 x^2], using Quiver plot it.
syms x y z
c(x,y)=curl([y^2 x^2 0],[x,y,z])
[X, Y] = meshgrid(-1:.1:1,-1:.1:1);
U=Y^2;
V=X^2;
quiver(X,Y,U,V)
%%
% b)
syms x y z
c(x,y)=curl([sin(pi*y) sin(pi*x) 0],[x,y,z])
[X, Y] = meshgrid(-1:.1:1,-1:.1:1);
U=sin(pi*Y);
V=sin(pi*X);
quiver(X,Y,U,V)
```

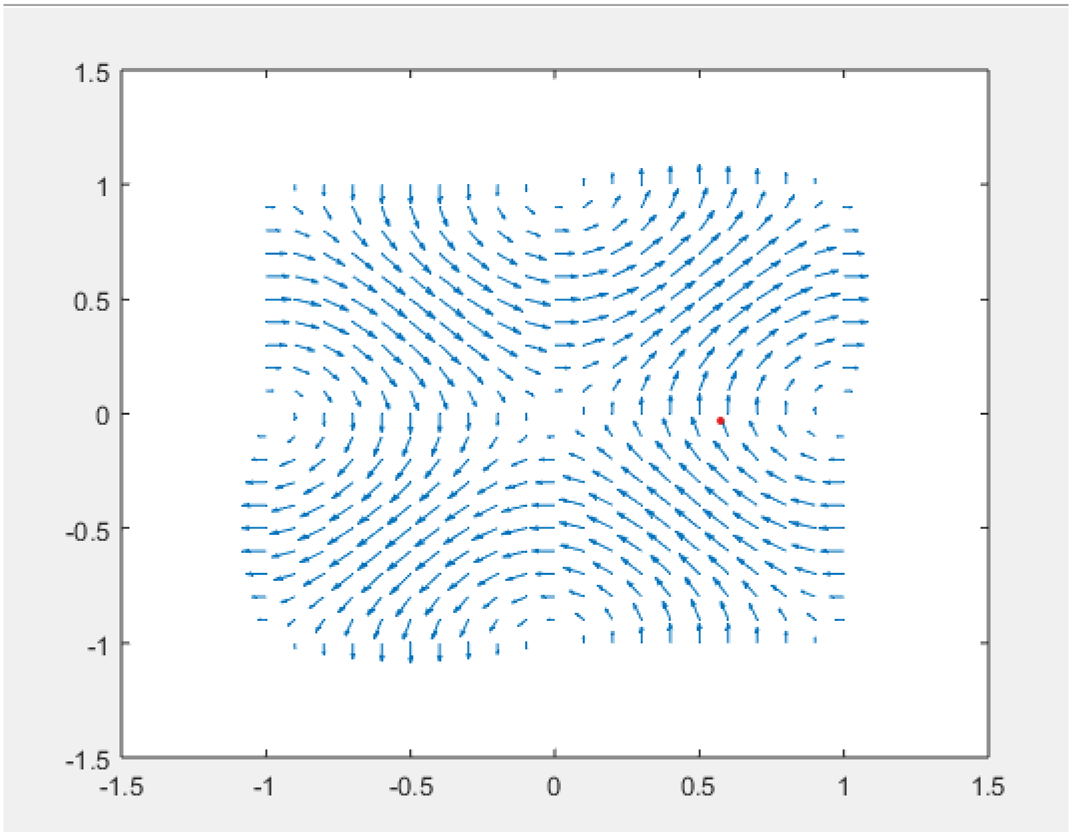
Output:-

```
c(x, y) =

         0
         0
 2*x - 2*y


c(x, y) =

                            0
                            0
 pi*cos(pi*x) - pi*cos(pi*y)
```


1.5
1
0.5
0
-0.5
-1
-1.5
-1.5
-1
-0.5
0
0.5
1
1.5