Name : Mudit Sand

Roll No. : 203100068

```
#Importing the libraries
import numpy as np
from sympy import *
import math
```

```
## Defining the qudratic function using sympy
x, y = symbols('x y')
func = 4*x**2 - 4*x*y + 2*y**2
func
```

$$4x^2 - 4xy + 2y^2$$

```
## Finding the gradient value
x0 , y0 = 2 , 3
def grad(fx,x0,y0):
  gradx = diff(fx , x)
  grady = diff(fx, y)
  gx = gradx.subs([(x,x0),(y,y0)])
  gy = grady.subs([(x,x0),(y,y0)])
  return [gx,gy]
```

```
z = grad(func,x0,y0)
a = z[0]**2 + z[1]**2
a = math.sqrt(a)
z= np.array(z)
z = z/a
print(z)
```

```
[0.707106781186547 0.707106781186547]
```

```
## Finding the function value = g
def g(func,x0,y0):
  gx = func.subs([(x,x0),(y,y0)])
  return gx
```

```
g1 = g(func,x0,y0)
```

```
##Steepest Descent
def steepest(func,x0,y0,N=100,tol=0.05):
  k=1
  while k < N :
    g1 = g(func,x0,y0)
    z = grad(func,x0,y0)
    a = z[0]**2 + z[1]**2
    z0 = math.sqrt(a)
    if z0 == 0:
      print("gradient zero")
      return x0,y0
    z = np.array(z)
    z = z/z0
    alp1 = 0
    alp3 = 1
    xn = x0 - alp3 * z[0]
    yn = y0 - alp3 * z[1]
    g3 = g(func,xn,yn)
    while g3>= g1:
      alp3 = alp3/2
      xn = x0 - alp3 * z[0]
      yn = y0 - alp3 * z[1]
      g3 = g(func,xn,yn)
      if alp3 <tol/2:
        print("No likely improvement")
        return x0,y0
    alp2 = alp3/2
    xn = x0 - alp2 * z[0]
    yn = y0 - alp2 * z[1]
    g2 = g(func,xn,yn)
    h1 = (g2-g1)/alp2
    h2 = (g3-g2)/(alp3-alp2)
```

```
    h3 = (h2-h1)/alp3
    alp0 = 0.5*(alp2 - (h1/h3))
    xn = x0 - alp0 * z[0]
    yn = y0 - alp0 * z[1]
    g0 = g(func,xn,yn)
    if g0 < g3:
      alp = alp0
      gfin = g0
    else:
      alp = alp3
      gfin = g3

    x0 = x0 - alp * z[0]
    y0 = y0 - alp * z[1]
    if abs(gfin-g1) < tol:
      print("success")
      return x0,y0
    k = k+1
  return print("unsuccess")
```

```
steepest(func,x0,y0)
```

```
     success
     (8.46545056276682e-16, 0.0400000000000059)
```

```
func2 = x**2 + y**2
steepest(func2,0,0)
```

```
     gradient zero
     (0, 0)
```