Set B

1. Check if the function is analytic or not & plot the same. [Also define the analytic function]

$f(x, y) = x^2 - y^2 + 2ixy$
$f(x, y) = sin(x)cosh(y) + icos(x)sinh(y)$

In [35]:

```
from sympy import *
import math
import cmath
import sympy as sy

def analytic(u,v):
    print("Given expression f(z):",(u+1j*v))
    diff_u_x=sy.diff(u,x)
    print("\nDerivative of u wrt x:",diff_u_x)
    diff_u_y=sy.diff(u,y)
    print("Derivative of u wrt y:",diff_u_y)
    diff_v_x=sy.diff(v,x)
    print("Derivative of v wrt x:",diff_v_x)
    diff_v_y=sy.diff(v,y)
    print("Derivative of v wrt y:",diff_v_y)
    if(diff_u_x == diff_v_y and diff_u_y == -diff_v_x):
        print("\nf(z) is an analytic function.")
        return True
    else:
        print("\nf(z) is not an analytic function.")
        return False

x = symbols('x')
y = symbols('y')
u = x**2 - y**2
v = 2*x*y
analytic(u,v)
```

```
Given expression f(z): x**2 + 2.0*I*x*y - y**2

Derivative of u wrt x: 2*x
Derivative of u wrt y: -2*y
Derivative of v wrt x: 2*y
Derivative of v wrt y: 2*x

f(z) is an analytic function.
```

Out[35]:

```
True
```

An analytic function is a function where the Cauchy Reimann equations are satisfied.

In [37]:

```
u = sin(x)*cos(y)
v = cos(x)*sin(y)
analytic(u,v)
```

```
Given expression f(z): sin(x)*cos(y) + 1.0*I*sin(y)*cos(x)

Derivative of u wrt x: cos(x)*cos(y)
Derivative of u wrt y: -sin(x)*sin(y)
Derivative of v wrt x: -sin(x)*sin(y)
Derivative of v wrt y: cos(x)*cos(y)

f(z) is not an analytic function.
```

Out[37]:

```
False
```

2. Plot 3 complex numbers on polar plane

In [27]:

```
def plotcomp(r1,r2,r3,i1,i2,i3):
    import matplotlib.pyplot as plt
    import cmath as cm
    import numpy as np
    z1 = complex(r1,i1)
    z2 = complex(r2,i2)
    z3 = complex(r3,i3)
    plt.polar([z1,z2,z3],marker="o")
    plt.show()

r1 = int(input("Enter real part of 1st complex number: "))
r2 = int(input("Enter real part of 2nd complex number: "))
r3 = int(input("Enter real part of 3rd complex number: "))
i1 = int(input("Enter imaginary part of 1st complex number: "))
i2 = int(input("Enter imaginary part of 2nd complex number: "))
i3 = int(input("Enter imaginary part of 3rd complex number: "))
plotcomp(r1,r2,r3,i1,i2,i3)
```

```
Enter real part of 1st complex number: 1
Enter real part of 2nd complex number: 2
Enter real part of 3rd complex number: 3
Enter imaginary part of 1st complex number: 4
Enter imaginary part of 2nd complex number: 5
Enter imaginary part of 3rd complex number: 6
```

```
C:\Users\Jeevan\Anaconda3\lib\site-packages\numpy\core\_asarray.py:85: Compl
exWarning: Casting complex values to real discards the imaginary part
  return array(a, dtype, copy=False, order=order)
```