

Chapter 5

Math

Chapter 5

Math

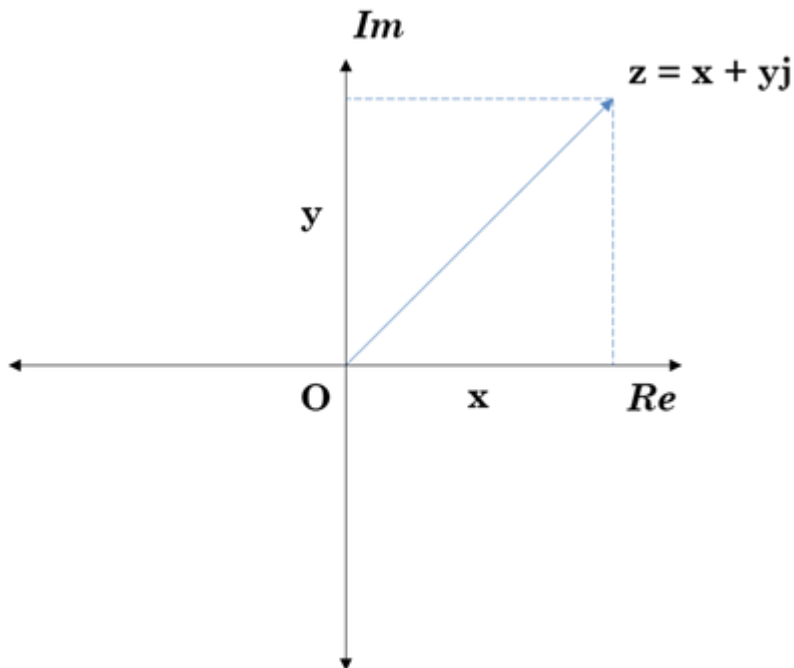
In This Chapter:

1. Polar Coordinates
2. Triangle Quest
3. Find Angle MBC

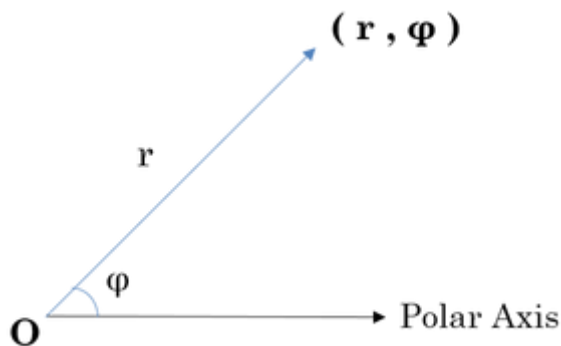
5.1 Polar Coordinates

Polar coordinates are an alternative way of representing Cartesian coordinates or Complex Numbers.

A complex number is z completely determined by its real part x and imaginary part y . Here, j is the imaginary unit.



A polar coordinate (r, φ) is completely determined by modulus r and phase angle φ .



If we convert complex number z to its polar coordinate, we find:

r : Distance from z to origin, i.e.,

$$\sqrt{x^2 + y^2}$$

φ : Counter clockwise angle measured from the positive x-axis to the line segment that joins z to the origin.

Python's **cmath** module provides access to the mathematical functions for complex numbers.

cmath.phase()

This tool returns the phase of complex number z (also known as the argument of z).

```
>>> phase(complex(-1.0, 0.0))  
3.1415926535897931
```

abs()

This tool returns the modulus (absolute value) of complex number .

```
>>> abs(complex(-1.0, 0.0))  
1.0
```

You are given a complex z . Your task is to convert it to polar coordinates.

Input Format

A single line containing the complex number z .

Output Format

Output two lines:

The first line should contain the value of r .

The second line should contain the value of ϕ .

Sample Input

```
1+2j
```

Sample Output

```
2.23606797749979
```

```
1.1071487177940904
```

Note: The output should be correct up to 3 decimal places.

Code

```
import cmath
data = eval(raw_input())
print abs(data)
print cmath.phase(data)
```

5.2 Triangle Quest

You are given a positive integer N .

Your task is to print a palindromic triangle of size N .

For example, a palindromic triangle of size 5 is:

```
1
121
12321
1234321
123454321
```

You can't take more than two lines. The first line (a *for*-statement) is already written for you.

You have to complete the code using exactly one print statement.

Note:

Using anything related to *strings* will give a score of 0.

Using more than one *for*-statement will give a score of 0.

Input Format

A single line of input containing the integer N .

Constraints

$$0 < N < 10$$

Output Format

Print the palindromic triangle of size N as explained above.

Sample Input

5

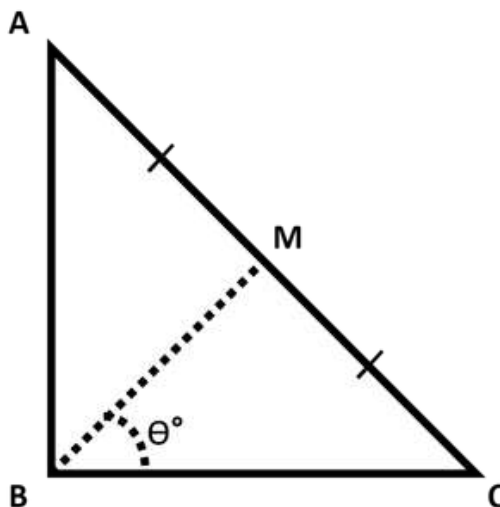
Sample Output

1
121
12321
1234321
123454321

Code

#More than 2 lines will result in 0 score.

```
for i in range(1,int(raw_input())+1):
    print((10**i-1)**2//81)
```

5.3 Find Angle MBC

ABC is a right triangle, at B.

Therefore,

$$\angle ABC = 90^\circ$$

Point M is the midpoint of hypotenuse AC .

You are given the lengths AB and BC .

Your task is to find $\angle MBC$ (angle , as shown in the figure) in degrees.

Input Format

The first line contains the length of side AB .

The second line contains the length of side BC .

Constraints

$$0 < AB \leq 100$$

$$0 < BC \leq 100$$

Lengths AB and BC are natural numbers.

Output Format

Output $\angle MBC^\circ$ in degrees.

Note: Round the angle to the nearest integer.

Examples:

If angle is 56.5000001° , then output **57** $^\circ$.

If angle is 56.5000000° , then output **57** $^\circ$.

If angle is 56.4999999° , then output **56** $^\circ$.

Sample Input

10

10

Sample Output

45 $^\circ$

Code

```
from __future__ import division
import math

ab = int(raw_input())
bc = int(raw_input())
c = math.degrees(math.atan(ab / bc))
c = int(round(c))
print str(c) + '°'
```