**Need to write the algorithm for the following**

1. **Biggest of 2 numbers**

**Step 1: Start**

**Step 2:** Take two numbers from the user as input, for say **n1** and **n2**.

**Step 3:** compare the two numbers using the **if** condition.

**Step 4:** if **n1** is greater than **n2**, then print **n1**.

**Step 5:** else print **n2**.

**Step 6: End**

1. **Biggest of 3 numbers**

**Step 1: Start**

**Step 2:** Take three numbers from the user as input, for say **n1**, **n2**, and **n3**.

**Step 3:** First compare whether **n1** is greater than **n2** and **n3** using the **if** condition.

**Step 4:** if the **Step 3** condition is **true** then print **n1** is greater.

**Step 5:** if the **Step 3** condition is **false** then compare whether **n2** is greater than **n1** and **n3** using the **if** condition.

**Step 6:** if the **Step 5** condition is **true** then print **n2** is greater.

**Step 7:** if both the **Step 3** and **Step 5** conditions are false, then print **n3** is greater.

**Step 8: End**

1. **Odd or Even**

Odd numbers are those numbers that cannot be divided into two equal parts.

Whereas even numbers are those numbers that can be divided into two equal parts

**Step 1: Start**

**Step 2:** Take one number from the user as input, for say **n1**.

**Step 3:** Use the **if** condition to check if the entered number is divisible by **2**.

**Step 4:** if the **Step 3** condition is **true**, then print the given number as **Even**

**Step 5:** if the **Step 3** condition is **false**, then print the given number as **Odd**

**Step 6: End**

1. **Prime or not**

A prime number is a natural number greater than 1 that has no positive divisor other than 1 and itself.

For example: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29

**Step 1: Start**

**Step 2:** Take one number from the user as input, for say **num1**.

**Step 3:** Use the **if** condition to check if the entered number is greater than **1**, if it is false then print please enter the number greater than 1.

**Step 4:** if the **Step 3** condition is **true**, then iterate from 2 to **num1** using a **for loop**.

**Step 5:** Use the **if** condition inside the for loop to check if the **num1** is divisible by any number between **2**.

**Step 6:** if the **Step 5** condition is **true** then print the entered number that is not prime number and break the for loop.

**Step 7:** if the **Step 5** condition is **false** then print the entered number as prime number.

**Step 8: End**

1. **Palindrome or not**

**Step 1: Start**

**Step 2:** Take a string from the user as input, for say **str1**.

**Step 3:** convert **str1** to suitable for caseless comparison using **casefold()** method.

**Step 4:** reverse the **str1** using the built-in function **reversed()** and store it in a new variable called **str2**.

**Step 5:** Use the **if** condition to compare whether **str1** is equal to **str2** by adding both strings to the list.

**Step 6:** if the **Step 5** condition is **true** then print the entered string as palindrome.

**Step 7:** if the **Step 5** condition is **false** then print the entered string is not palindrome.

**Step 8: End**

**Note: casefold() method returns a lowercase version of the string.**

**Note: we use list() function to convert them into a list before comparing.**

1. **Quadratic equation**

**Step 1: Start**

**Step 2: import math**

**Step 3:** Take three numbers from the user as input, for say **n1**, **n2**, and **n3**.

**Step 4:** calculate the discriminant and store it in the variable called **dis**.

**Step 5:** Use the **if** condition to check whether **dis** is greater than zero, if **true** then print there are 2 roots.

**Step 6:** if the **Step 5** condition is **false** then check whether **dis** is equal to 0, if **true** then print there is one root.

**Step 7:** else print **no roots discriminant < 0**.

**Step 8: End**

**Convert the following into the appropriate format**

**a² – b² = (a-b)(a+b)**

(a\*a) - (b\*b) = (a-b)\*(a+b)

**(a+b)² = a² + 2ab + b²**

(a+b)\*(a+b) = (a\*a) + (2\*a\*b) + (b\*b)

**(a-b)² = a² – 2ab + b²**

(a-b)\*(a-b) = (a\*a) - (2\*a\*b) + (b\*b)

**a² + b² = (a-b)² +2ab**

(a\*a) + (b\*b) = (a-b)\*(a-b) + (2\*a\*b)

**(a+b+c)² = a²+b²+c²+2ab+2ac+2bc**

(a+b+c)\*(a+b+c) = (a\*a) + (b\*b) + (c\*c) + (2\*a\*b) + (2\*a\*c) + (2\*b\*c)

**(a-b-c)² = a²+b²+c²-2ab-2ac+2bc**

(a-b-c)\*(a-b-c) = (a\*a) + (b\*b) + (c\*c) - (2\*a\*b) - (2\*a\*c) + (2\*b\*c)

**a³-b³ = (a-b) (a² + ab + b²)**

(a\*a\*a) - (b\*b\*b) = (a-b) \* ((a\*a) + ab + (b\*b))

**a³+b³ = (a+b) (a² – ab + b²)**

(a\*a\*a) + (b\*b\*b) = (a+b) \* ((a\*a) - ab + (b\*b))

**(a+b)³ = a³+ 3a²b + 3ab² + b³**

(a+b)\*(a+b)\*(a+b) = (a\*a\*a) + (3\*(a\*a)\*b) + (3\*a\*(b\*b)) + (b\*b\*b)

**(a-b)³ = a³- 3a²b + 3ab² – b³**

(a+b)\*(a+b)\*(a+b) = (a\*a\*a) - (3\*(a\*a)\*b) + (3\*a\*(b\*b)) - (b\*b\*b)

**IN SIMPLE**

——————

a² → (a\*a)

b² → (b\*b)  
a³ → (a\*a\*a)

b³ → (b\*b\*b)

(a+b)² → (a+b) \* (a\*b)

(a-b)² → (a-b) \* (a-b)

(a+b)³ → (a+b)\*(a+b)\*(a+b)

(a-b)³ → (a-b)\*(a-b)\*(a-b)

(a+b+c)² → (a+b+c)\*(a+b+c)

(a-b-c)² → (a-b-c)\*(a-b-c)

a² + b² → (a\*a) + (b\*b)

a² - b² → (a\*a) - (b\*b)

a³-b³ → (a\*a\*a) - (b\*b\*b)

a³+b³ → (a\*a\*a) + (b\*b\*b)

ab → (a\*b)

2ab → (2\*a\*b)

3a²b → (3\*(a\*a)\*b)

3ab² → (3\*a\*(b\*b))

**convert the above to the Python syntax-based**

**a = 5**

**b = 2**

**c = 5**

**# (a+b)² = a² + 2ab + b²**

**# d = a\*a+2\*a\*b+b\*b == (a\*a) + (2\*a\*b) + (b\*b)**

**print("Equation 1")**

**d1 = (a+b)\*\*2**

**d2 = (a\*\*2) + (2\*a\*b) + (b\*\*2)**

**d3 = d1 == d2**

**print(d1)**

**print(d2)**

**print(d3)**

**if d1 and d2:**

**print("LHS = RHS")**

**print("-" \* 10)**

**# (a-b)² = a² – 2ab + b²**

**# e = (a-b)\*\*2 == (a\*a) - (2\*a\*b) + (b\*b)**

**print("Equation 2")**

**e1 = (a-b)\*\*2**

**e2 = (a\*\*2) - (2\*a\*b) + (b\*\*2)**

**e3 = e1 == e2**

**print(e1)**

**print(e2)**

**print(e3)**

**if e1 and e2:**

**print("LHS = RHS")**

**print("-" \* 10)**

**# a² + b² = (a-b)² +2ab**

**print("Equation 3")**

**f1 = (a\*\*2) + (b\*\*2)**

**f2 = (a-b)\*\*2 + (2\*a\*b)**

**f3 = f1 == f2**

**print(f1)**

**print(f2)**

**print(f3)**

**if f1 and f2:**

**print("LHS = RHS")**

**print("-" \* 10)**

**# (a+b+c)² = a²+b²+c²+2ab+2ac+2bc**

**# g = (a+b+c)\*\*2 == (a\*\*2) + (b\*\*2) + (c\*\*2) + (2\*a\*b) + (2\*a\*c) + (2\*b\*c) // Error**

**print("Equation 4")**

**g1 = (a+b+c)\*\*2**

**g2 = (a\*\*2) + (b\*\*2) + (c\*\*2) + (2\*a\*b) + (2\*a\*c) + (2\*b\*c)**

**g3 = g1 == g2**

**print(g1)**

**print(g2)**

**print(g3)**

**if g1 and g2:**

**print("LHS = RHS")**

**print("-" \* 10)**

**# (a-b-c)² = a²+b²+c²-2ab-2ac+2bc**

**print("Equation 5")**

**h1 = (a-b-c)\*\*2**

**h2 = (a\*\*2) + (b\*\*2) + (c\*\*2) - (2\*a\*b) - (2\*a\*c) + (2\*b\*c)**

**h3 = h1 == h2**

**print(h1)**

**print(h2)**

**print(h3)**

**if h1 and h2:**

**print("LHS = RHS")**

**print("-" \* 10)**

**# a³-b³ = (a-b) (a² + ab + b²)**

**print("Equation 6")**

**i1 = (a\*\*3) - (b\*\*3)**

**i2 = (a-b) \* ((a\*\*2) + (a\*b) + (b\*\*2))**

**i3 = i1 == i2**

**print(i1)**

**print(i2)**

**print(i3)**

**if i1 and i2:**

**print("LHS = RHS")**

**print("-" \* 10)**

**# a³+b³ = (a+b) (a² – ab + b²)**

**print("Equation 7")**

**j1 = (a\*\*3) + (b\*\*3)**

**j2 = (a+b) \* ((a\*\*2) - (a\*b) + (b\*\*2))**

**j3 = j1 == j2**

**print(j1)**

**print(j2)**

**print(j3)**

**if j1 and j2:**

**print("LHS = RHS")**

**print("-" \* 10)**

**# (a+b)³ = a³+ 3a²b + 3ab² + b³**

**# k = (a+b)\*\*3 == (a\*\*3) + (b\*\*3) + ((3\*a\*b) \* (a+b))**

**print("Equation 8")**

**k1 = (a+b)\*\*3**

**k2 = (a\*\*3) + ((3\*(a\*\*2)\*b)) + (3\*a\*(b\*\*2)) + (b\*\*3)**

**k3 = k1 == k2**

**print(k1)**

**print(k2)**

**print(k3)**

**if k1 and k2:**

**print("LHS = RHS")**

**print("-" \* 10)**

**# (a-b)³ = a³- 3a²b + 3ab² – b³**

**print("Equation 9")**

**l1 = (a-b)\*\*3**

**l2 = (a\*\*3) - ((3\*(a\*\*2)\*b)) + (3\*a\*(b\*\*2)) - (b\*\*3)**

**l3 = l1 == l2**

**print(l1)**

**print(l2)**

**print(l3)**

**if l1 and l2:**

**print("LHS = RHS")**

**print("-" \* 10)**

**# a² – b² = (a-b)(a+b)**

**print("Equation 10")**

**m1 = (a\*\*2) - (b\*\*2)**

**m2 = (a-b) \* (a+b)**

**m3 = m1 == m2**

**print(m1)**

**print(m2)**

**print(m3)**

**if m1 and m2:**

**print("LHS = RHS")**

**print("-" \* 10)**

**OUTPUT**

**nmadhu@BAN-LAP-NMADHU ~ % /usr/bin/python3 /Users/nmadhu/test.py**

**Equation 1**

49

49

True

LHS = RHS

**----------**

**Equation 2**

9

9

True

LHS = RHS

**----------**

**Equation 3**

29

29

True

LHS = RHS

**----------**

**Equation 4**

144

144

True

LHS = RHS

**----------**

**Equation 5**

4

4

True

LHS = RHS

**----------**

**Equation 6**

117

117

True

LHS = RHS

**----------**

**Equation 7**

133

133

True

LHS = RHS

**----------**

**Equation 8**

343

343

True

LHS = RHS

**----------**

**Equation 9**

27

27

True

LHS = RHS

**----------**

**Equation 10**

21

21

True

LHS = RHS

**----------**