**Python Program for Number of solutions to Modular Equations**

**import math**

**def calculateDivisors (A, B):**

**N = A - B**

**noOfDivisors = 0**

**a = math.sqrt(N)**

**for i in range(1, int(a + 1)):**

**if ((N % i == 0)):**

**if (i > B):**

**noOfDivisors +=1**

**if ((N / i) != i and (N / i) > B):**

**noOfDivisors += 1;**

**return noOfDivisors**

**def numberOfPossibleWaysUtil (A, B):**

**if (A == B):**

**return -1**

**if (A < B):**

**return 0**

**noOfDivisors = 0**

**noOfDivisors = calculateDivisors;**

**return noOfDivisors**

**def numberOfPossibleWays(A, B):**

**noOfSolutions = numberOfPossibleWaysUtil(A, B)**

**if (noOfSolutions == -1):**

**print ("For A = " , A , " and B = " , B , ", X can take Infinitely many values", " greater than " , A)**

**else:**

**print ("For A = " , A , " and B = " , B , ", X can take " , noOfSolutions , " values")**

**A = 26**

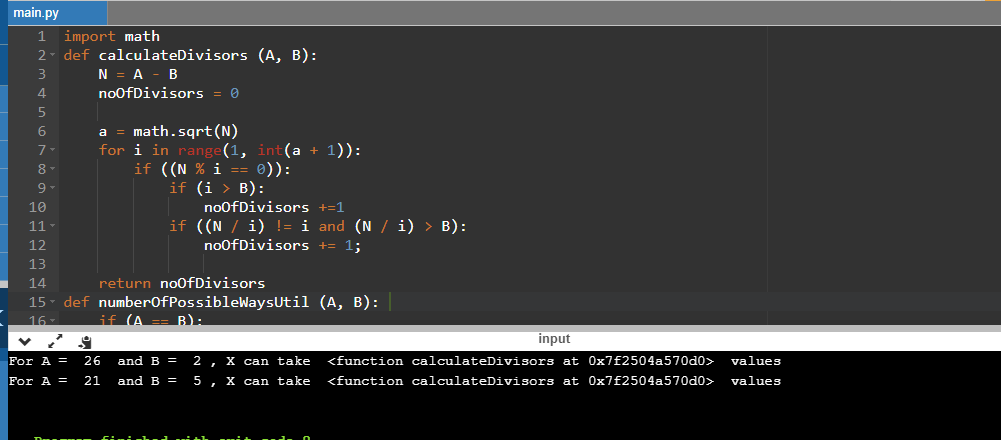
**B = 2**

**numberOfPossibleWays(A, B)**

**A = 21**

**B = 5**

**numberOfPossibleWays(A, B)**

****