Российский университет дружбы народов Научный факультет

Математические основы защиты информации и

информационной безопасности

МСД

НФИмд 02-22

Подготовлено студентом:

Елиенис Санчес Родригес.

Преподаватель: Дмитрий Сергеевич

Что такое наибольший общий делитель (MCD)

В математике наибольший общий делитель или MCD называется наибольшим числом, которое делит ровно два или более чисел одновременно. Поскольку мы говорим о наибольшем числе, мы будем принимать во внимание только положительные делители.

Мы также можем сказать, что наибольший общий делитель двух чисел "А «и» В" - это большее число, которое делит их на два, как на число А, так и на число В.

Por Например, мы скажем, что наибольший общий делитель 18 и 24 равен 6, потому что 6-это наибольший из общих делителей 18 и 24, и запишем его как MCD (18,24) = 6

Учитываются числа, на которые деления дают нулевой остаток. Вы можете просмотреть деления на одну цифру, если хотите, чтобы помочь вам запомнить части, составляющие деление.

Для чего используется наибольший общий делитель

MCD для упрощения дробей

Одна из утилит, которые имеет наибольший общий делитель, - это упрощение дробей.

Например, чтобы упростить дробь 12/18, сначала вычисляется Наибольший Общий делитель 12 и 18, равный 6.

Затем мы должны разделить числитель и знаменатель начальной дроби на 6, чтобы получить упрощенную дробь, равную 2/3.

MCD для вычисления наименьшего общего кратного (mcm)

Наибольший общий делитель также можно использовать для вычисления наименьшего общего, кратного двум числам, их mcm.

Это потому, что произведение наибольшего общего делителя двух чисел на наименьшее общее кратное (тех же чисел) равно произведению этих двух чисел.

Давайте посмотрим на это на примере. Как мы уже говорили ранее, MCD (12,18) = 6 как 12 × 18 = 216, его наименьшее общее кратное должно быть 36, потому что 6 × 36 = 216.

MCD для решения проблем

Без сомнения, больше всего вы собираетесь использовать MCD для решения проблем.

наибольший общий делитель-это наибольшее число среди общих делителей.

Давайте посмотрим, каков наибольший общий делитель из приведенного выше примера, MCD (15,20).

Делители 15: 1, 3, 5 и 15.

Делители 20: 1, 2, 4, 5, 10 и 20.

Вы уже знаете, что общие делители на 15 и 20 равны 1 и 5, теперь из этих двух чисел (1 и 5) вы должны выбрать большее число, которое равно 5. Таким образом, наибольший общий делитель на 15 и 20 равен 5.

Как сделать наибольший общий делитель

Чтобы узнать, каков наибольший общий делитель двух или более чисел, существует несколько методов, давайте рассмотрим два.

**Метод 1 расчета MCD**

Для этого важно очень хорошо понять, что означает наибольший общий делитель.

Мы записываем все делители каждого числа и из них указываем общие делители.

Наибольшим делителем будет MCD этих чисел.

Недостатком этого метода является то, что большое число может иметь много делителей, и запись их всех может быть очень громоздкой. Преимущество этого в том, что если вы это учтете, иногда вам не придется ничего вычислять, обратите внимание, если вас спросят о наибольшем общем делителе из трех чисел, и окажется, что одно делит другие, у вас уже есть наибольший общий делитель. Например, наибольший общий делитель на 36, 12 и 84. Поскольку 12 делит все три, не может быть большего общего делителя.

**Метод 2 расчета MCD**

Разложение множителей или разложение на простые числа.

Мы разлагаем каждое число на простые множители.

Затем мы указываем на общие факторы.

Затем в каждом из общих значений мы выбираем коэффициент с наименьшим показателем степени.

И, наконец,умножаем выбранные факторы.

**Метод 3 Алгоритм Евклида**

Алгоритм Евклида - это**процедура вычисления наибольшего общего делителя (m.c.d.)**двух чисел.

Евклид был греческим математиком, который собрал различные данные в работе под названием Элементы. Эта работа считается одним из столпов математики, а Евклид - "отцом геометрии".

В разделе "Элементы" Евклид объясняет, что наибольший общий делитель двух чисел можно найти, разделив большее число на меньшее. Если деление точное, m.c.d. наименьшее число. Если деление неточное, то берется остаток и делится столько раз, сколько потребуется, чтобы получить деление без остатка. M. c. d. - последнее число, на которое его можно разделить.

Хотя слово "алгоритм" заставляет нас думать о сложных вычислениях, решаемых компьютерами, в нашем случае вычисления намного проще. Все, что вам нужно сделать, это выполнить следующие шаги

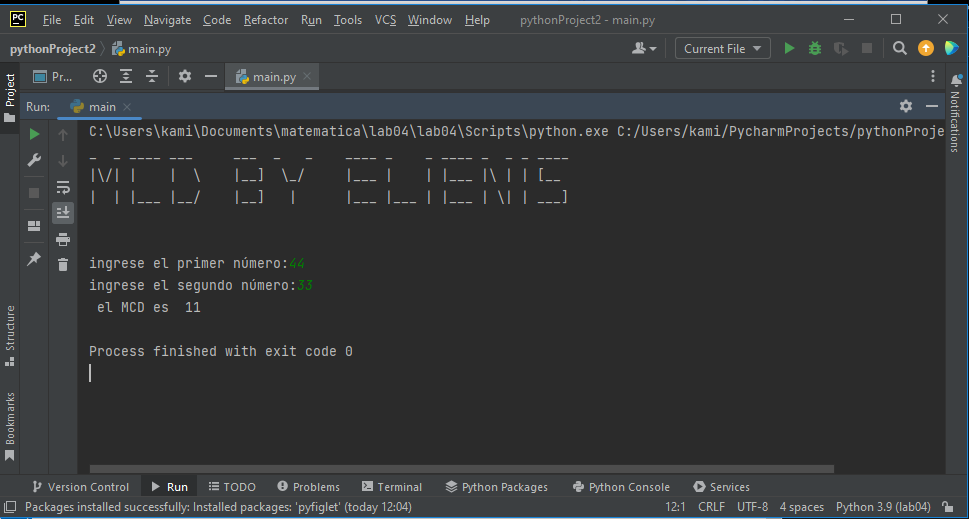
1. Большее число делится на меньшее.
2. Если деление точное, делителем является m. c. d.
3. Если деление неточное, мы делим делитель на полученный остаток и продолжаем в том же духе, пока не получим точное деление. M. c. d. является последним делителем.

Расширенный алгоритм Евклида представляет собой небольшую модификацию, которая позволяет дополнительно выразить наибольший общий делитель как линейную комбинацию. Этот алгоритм находит применение, в частности, в различных областях, таких как алгебра, теория чисел и информатика. С небольшими модификациями он обычно используется в электронных компьютерах из-за его высокой эффективности.

В вычислениях с двоичной системой счисления деление и умножение на 2 сводятся к перемещению места справа или слева от умножаемой цифры. По этой причине интересно выделить число 2 в расчетах. Первая идея состоит в том, что если мы выразим два числа**А**и**B**В в виде**А=2m\*p**,**B=2n\*q**, где p и q являются наибольшими нечетными делителями, M. C. D можно найти с помощью двух отдельных вычислений. С одной стороны,мы остаемся с наименьшим показателем степени 2, а с другой стороны, находим MCD(p, q).

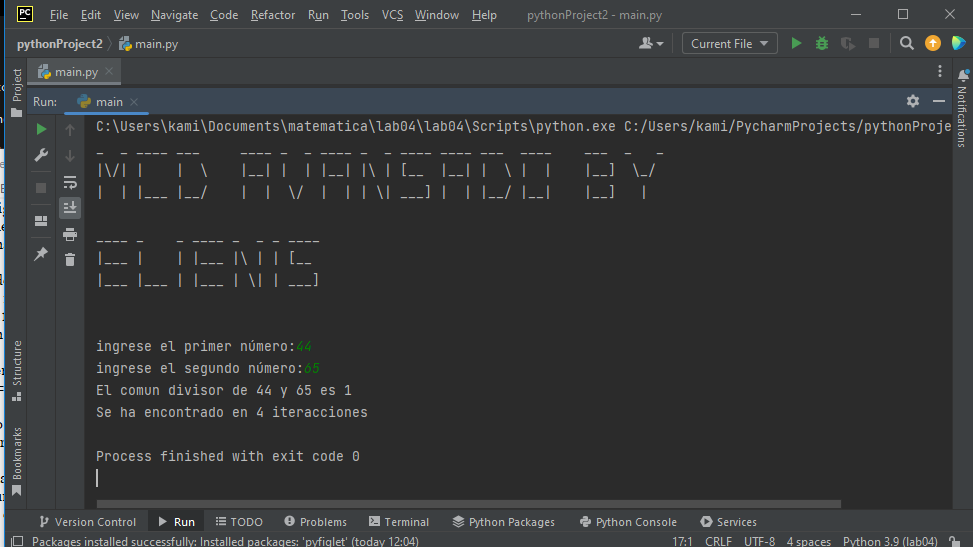
В двоичном алгоритме Евклидастепень 2, общая для обоих чисел, сначала удаляется и принимается к сведению. После удаления фактор 2 не повлияет на результат, и когда он появится в любом из двух чисел, его также можно будет удалить. В двоичном выражении удаление 2 означает перемещение цифр на одну позицию вправо.

from pyfiglet import figlet\_format  
print(figlet\_format( "MCD by Elienis", font = "cybermedium"))  
import math  
  
comprobar = True  
  
while comprobar:  
 #introducimos las variables en numeros enteros  
 a = int(input("ingrese el primer número:"))  
 b = int(input("ingrese el segundo número:"))  
  
 MCD = False  
 # si a y b es mayor que 0 y que a y b sean diferentes  
 if a > 0 and b > 0 and a != b:  
  
 comprobar = False  
 #creo variable auxiliar en caso de que b sea menor que a  
 if b < a:  
 aux = a  
 a=b  
 b= aux  
  
 i=a  
 #creamos ciclo mientras mcd sea falso y i sea mayor o igual a 1  
 while not MCD and i >= 1:  
 #si a es igual a 0 e i , imprimimos esa variable  
 if a % i == 0 and b % i== 0 :  
 print(" el MCD es ",i)  
 MCD =True  
  
 #decrementamos 1 en el bucle hasta que i sea una division exacta  
 else:  
 i-= 1  
  
 # si el usuario no coloco los numeros correctos , manda a pedir los numeros  
 else:  
 if a == b:  
 print("Los numeros son iguales, intentelo de nuevo ")  
 else:  
 print("Los numeros no son correctos, intentelo neuvamente")



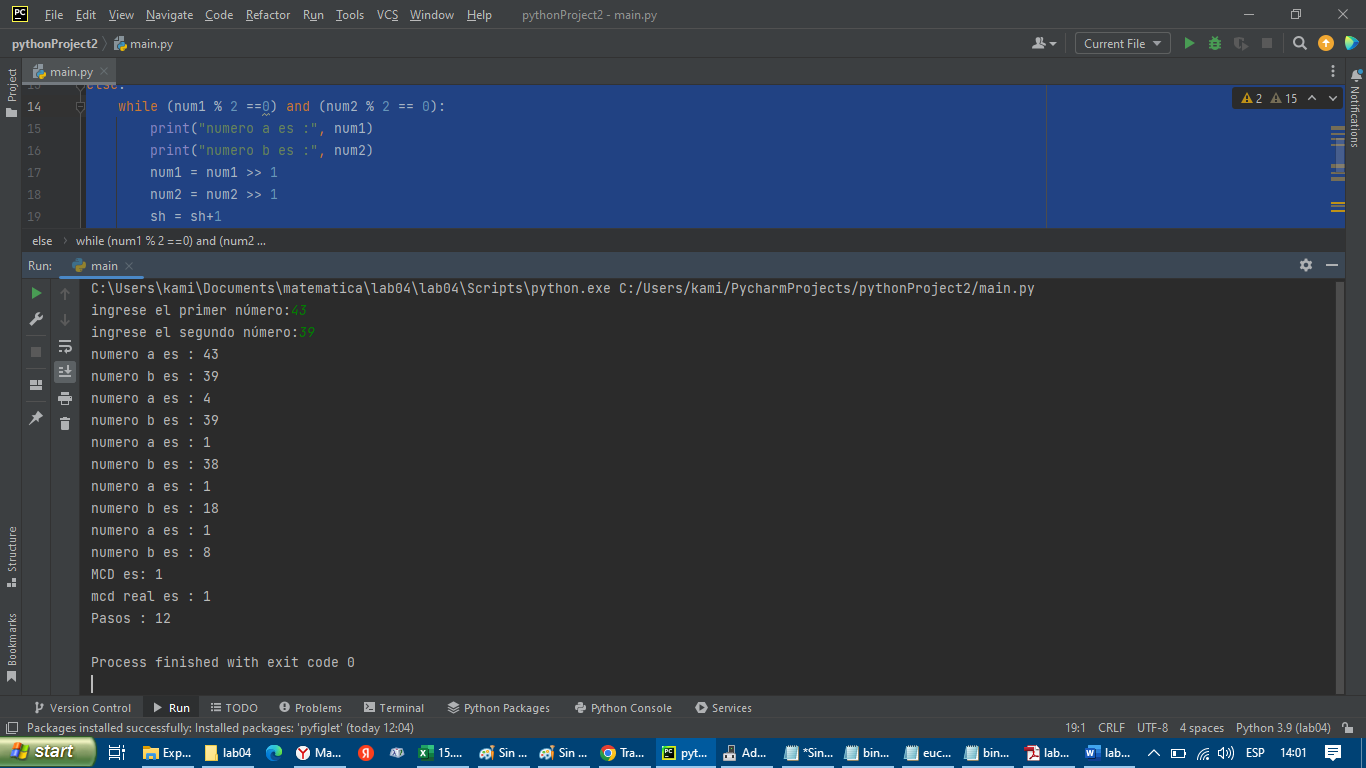
2.

from pyfiglet import figlet\_format  
print(figlet\_format( "MCD Avansado by Elienis", font = "cybermedium"))  
import math  
  
def euclides(num1, num2, iteracciones=1):  
 # Si el num1 es inferior al num2, los invertimos  
 if num1 < num2:  
 num1, num2 = num2, num1  
  
 # obtenemos el resto de la division  
 resto = num1 % num2  
  
 if resto == 0:  
 return (num2, iteracciones)  
  
 # llamamos nuevamente a la función pasando como primer parametro el  
 # segundo numero y el resto de la division  
 return euclides(num2, resto, iteracciones + 1)  
  
a = int(input("ingrese el primer número:"))  
b = int(input("ingrese el segundo número:"))  
num1 = a  
num2 = b  
  
comunDivisor, iteracciones = euclides(num1, num2)  
  
print("El comun divisor de {} y {} es {}".format(num1, num2, comunDivisor))  
print("Se ha encontrado en {} iteracciones".format(iteracciones))



3.

import math  
  
a = int(input("ingrese el primer número:"))  
b = int(input("ingrese el segundo número:"))  
num1 = a  
num2 = b  
sh=0  
steps=0  
  
if (num1 == 0) or (num2 ==0):  
 ans = 0  
 #si la respuesta es 0 no necesitamos continuar  
else:  
 while (num1 % 2 ==0) and (num2 % 2 == 0):  
 print("numero a es :", num1)  
 print("numero b es :", num2)  
 num1 = num1 >> 1  
 num2 = num2 >> 1  
 sh = sh+1  
 steps = steps+1  
  
 while (num1 != num2):  
 print("numero a es :",num1)  
 print("numero b es :", num2)  
 while (num1 & 1 ==0):  
 num1 = num1 >> 1  
 steps = steps +1  
 while (num2 & 1 ==0):  
 num2 = num2 >> 1  
 steps = steps + 1  
 steps = steps + 1  
 if num1 < num2:  
 num2 = num2 - num1  
 if num2 < num1:  
 num1 = num1 - num2  
 ans = num1 << sh  
print("MCD es:" ,ans)  
print("mcd real es :", math.gcd(a,b))  
print("Pasos :", steps)  
5



4.

def extended\_gcd(a, b):  
 if a == 0:  
 return b, 0, 1  
 else:  
 gcd, x, y = extended\_gcd(b % a, a)  
 return gcd, y - (b // a) \* x, x  
  
a = int(input("ingrese el primer número:"))  
b = int(input("ingrese el segundo número:"))  
x = a  
y = b  
if \_\_name\_\_ == '\_\_main\_\_':  
 gcd, x, y = extended\_gcd(x, y)  
 print('el maximo comun divisor es', gcd)  
  
 print("los coeficientes de la identidad de Bézout son")  
 print(f'x = {x}, y = {y}')

