«Функции для нахождения делителей числа. Задача 25 ЕГЭ по информатике»

**Алгоритмы с оптимизацией**

**Функция для подсчета количества делителей числа**

def dell(x):

k = 0

for d in range(1, int(x\*\*0.5)+1):

if x % d == 0:

if d != x//d:

k += 2

else:

k += 1

return k

print(dell(100000000))

**Функция для проверки простоты числа**

def prime(x):

if x == 1: return False

for d in range(2, int(x\*\*0.5)+1):

if x % d == 0:

return False

return True

print(prime(2))

**Функция, возвращающая все делители числа**

**1-ый вариант**

def dell(x):

dd = []

for d in range(1, int(x\*\*0.5)+1):

if x % d == 0:

dd.append(d)

if d != x//d:

dd.append(x//d)

return sorted(dd)

print(dell(24))

2-ой вариант

def getdel(x):

dd = set()

for d in range(1, int(x\*\*0.5)+1):

if x % d == 0:

dd.add(d)

dd.add(x//d)

return sorted(dd)

print(dell(24))

Задачи с сайта К.Полякова

**Р-01**. Напишите программу, которая ищет среди целых чисел, принадлежащих числовому отрезку [3532000; 3532160], простые числа. Выведите все найденные простые числа в порядке возрастания, слева от каждого числа выведите его номер по порядку.

def prime(x):

if x == 1: return False

for d in range(2, int(x\*\*0.5)+1):

if x % d == 0:

return False

return True

k = 1

for i in range(3532000, 3532160+1):

if prime(i):

print(k, i)

k += 1

70.(**Д.Ф. Муфаззалов, г. Уфа**) Совершенным называется число, [натуральное число](https://ru.wikipedia.org/wiki/%D0%9D%D0%B0%D1%82%D1%83%D1%80%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D0%B5_%D1%87%D0%B8%D1%81%D0%BB%D0%BE), равное сумме всех своих [собственных делителей](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D0%B1%D1%81%D1%82%D0%B2%D0%B5%D0%BD%D0%BD%D1%8B%D0%B9_%D0%B4%D0%B5%D0%BB%D0%B8%D1%82%D0%B5%D0%BB%D1%8C) (то есть всех положительных делителей, отличных от самого́ числа) (например, число 6=1+2+3). ) Выведите каждое совершенное число из диапазона [2; 10000] и количество его собственных делителей в порядке возрастания. Вывод каждого совершенного числа начинайте с новой строки. Числа в строке разделяйте пробелом.

def dell(x):  
 s = set()  
 sm = 1  
 for k in range(2,int(x \*\* 0.5)+1):  
 if x % k == 0:  
 sm += k + x//k  
 s.add(k)  
 s.add(x//k)  
 return sm,sorted(s)  
for i in range(2,10000):  
 sm, b = dell(i)  
 if i == sm:  
 print(i,len(b)+1)

38.(**Б.С. Михлин**) Напишите программу, которая ищет среди целых чисел, принадлежащих числовому отрезку [268220; 270335] число c максимальной суммой делителей, имеющее не более четырех делителей. Для найденного числа выведите сумму делителей, количество делителей и все делители в порядке убывания.

def dell(x):  
 s = set()  
 for k in range(1,int(x\*\*0.5)+1):  
 if x % k == 0:  
 s.add(k)  
 s.add(x//k)  
 return sorted(s,reverse=True)  
mx = 0  
for i in range(268220,270335+1):  
 if len(dell(i)) <= 4:  
 b = list(dell(i))  
 if sum(b) > mx:  
 mx = sum(b)  
 d = b  
print(mx,len(d),\*d,sep= " ")

110.Среди целых чисел, принадлежащих числовому отрезку [237981; 309876], найдите числа, которые представляют собой произведение двух различных простых делителей, заканчивающихся на одну и ту же цифру. Запишите в ответе количество таких чисел и максимальное их них.

def dell(x):  
 s = set()  
 for k in range(1,int(x\*\*0.5)+1):  
 if x % k == 0:  
 s.add(k)  
 s.add(x//k)  
 return s  
def prime(x):  
 if x == 1: return False  
 for k in range(2,int(x\*\*0.5)+1):  
 if x % k == 0: return False  
 return True  
d = []  
for i in range(237981,309876+1):  
 b = [x for x in dell(i) if prime(x)]  
 if len(b) == 2:  
 if b[0] % 10 == b[1] % 10 and b[0] \* b[1] == i:  
 d.append(i)  
print(len(d),max(d))

**Сложные задачи с делителями**

138.Рассмотрим произвольное натуральное число, представим его всеми возможными способами в виде произведения двух натуральных чисел и найдём для каждого такого произведения разность сомножителей. Например, для числа 18 получим: 18 = 18\*1 = 9\*2 = 6\*3, множество разностей содержит числа 17, 7 и 3. Подходящей будем называть пару сомножителей, разность между которыми не превышает 110. Найдите все натуральные числа, принадлежащие отрезку [1000000; 1500000], у которых есть не менее трёх подходящих пар сомножителей. В ответе перечислите найденные числа в порядке возрастания, справа от каждого запишите наибольший из всех сомножителей, образующих подходящие пары.

def dell(x):  
 s = set()  
 for k in range(1,int(x\*\*0.5)+1):  
 if x % k == 0:  
 if x // k - k <= 110:  
 s.add(k)  
 s.add(x//k)  
 return sorted(s)  
for i in range(1000000,1500000+1):  
 sm = dell(i)  
 if len(sm) >=6:  
 print(i, sm[-1])

186.Обозначим через P(N) – произведение 5 наименьших различных нетривиальных делителей натурального числа N (не считая единицы и самого числа). Если у числа N меньше 5 таких делителей, то P(N) считается равным нулю. Найдите 5 наименьших натуральных чисел, превышающих 500 000 000, для которых P(N) оканчивается на 91 и не превышает N. В ответе для каждого найденного числа запишите сначала значение P(N), а затем – наибольший делитель, вошедший в произведение P(N).

def dell(x):  
 s1 = set()  
 s = 1  
 for k in range(2,int(x\*\*0.5)+1):  
 if x % k == 0:  
 s1.add(k)  
 s = s \* k  
 if len(s1) == 5:return sorted(s1), s  
 return 0,0  
for i in range(500\_000\_000, 500\_000\_000 + 10 \*\* 10):  
 d,p = dell(i)  
 if p % 100 == 91 and len(d) == 5 and p < i:  
 print(p,d[-1])

**Формула вычисления количества делителей числа через известное его разложение на простые множители**

Пусть X= d1⋅p1⋅d2⋅p2⋅…..⋅dn⋅pn- разложение числа X на простые множители. Здесь di- простые числа, pi - натуральные числа. Например, 36=22⋅32.

Тогда количество делителей числа X может быть вычислено по формуле (p1+1)⋅(p2+1)⋅⋯⋅(pn+1)(p1+1)⋅(p2+1)⋅⋯⋅(pn+1). Например, число делителей у числа 36 равно (2+1)(2+1)=9.

134.Назовём нетривиальным делителем натурального числа его делитель, не равный единице и самому числу. Найдите все натуральные числа, принадлежащие отрезку [525784203; 728943762] и имеющие ровно три нетривиальных делителя. Для каждого найденного числа запишите в ответе само число и его наибольший нетривиальный делитель. Найденные числа расположите в порядке возрастания.

def prime(x):  
 if x == 1: return False  
 for k in range(2,int(x \*\* 0.5)+1):  
 if x % k == 0: return False  
 return True  
for i in range(int(525784203\*\*0.25)+1,int(728943762\*\*0.25)+1):  
 if prime(i):  
 print(i\*\*4, i\*\*3)

148.Найдите все натуральные числа, принадлежащие отрезку [113 000 000; 114 000 000], у которых ровно три различных чётных делителя. В ответе перечислите найденные числа в порядке возрастания, справа от каждого числа запишите его второй по величине нетривиальный делитель (не равный 1 и самому числу). Первым по величине считается меньший из нетривиальных делителей числа.

def prime(x):  
 if x == 1: return False  
 for k in range(2,int(x \*\* 0.5)+1):  
 if x % k == 0: return False  
 return True  
  
print((113000000/2)\*\*0.5, (114000000/2)\*\*0.5)  
  
for p in range(7517, 7549+1):  
 if prime(p):  
 print(2\*p\*\*2, p)

164.Найдите все натуральные числа, N, принадлежащие отрезку [100 000 000; 300 000 000], которые можно представить в виде N = 2m · 5n, где m – нечётное число, n – чётное число. В ответе запишите все найденные числа в порядке возрастания, а справа от каждого числа – сумму m+n.

r = []  
for m in range(1,33,2):  
 for n in range(0,15,2):  
 N = 2\*\*m\*5\*\*n  
 if 100000000 <= N<= 300000000:  
 r.append((N,m + n))  
for x in sorted(r):  
 print(\*x)  
print(\*sorted(r))