

```

1 class divisor:
2     primes = [2,3,5,7,11,13] # list of prime numbers
3
4     def __init__(self, data):
5         self.data = data
6         self.divi = 0 # index in primes[]
7
8     def __iter__(self):
9         return self.next()
10
11    def next(self):
12        while True:
13            div = divisor.primes[self.divi]
14            while self.data % div == 0:
15                self.data /= div
16                yield div
17            if div * div > self.data:
18                break
19            self.divi += 1 # next divisor
20            if len(divisor.primes) <= self.divi:
21                newdiv = div + 2
22                while not prime(newdiv):
23                    newdiv += 2
24                divisor.primes.append(newdiv)
25            if self.data > 1:
26                yield self.data
27
28    def prime(n):
29        nb = 0
30        for d in divisor(n):
31            nb += 1
32            if nb > 1: return False
33        return True # n has only one divisor
34
35    while True:
36        try:
37            n = int(raw_input("Entrez un nombre : "))
38        except:
39            break
40        else:
41            if n < 0: n = -n
42            print "Diviseurs de ", n
43            for d in divisor(n):
44                print d

```

## TEXT STATISTICS

1175 characters  
44 lines

## LEXICAL STATISTICS

4 comments  
2 constants [string]  
16 constants [numeric]

## SYMBOL TABLE

False	32								
True	12	33	35						
__init__	4								
__iter__	8								
append	24								
d	30	43	44						
data	4	5+	14	15	17	25	26		
div	13	14	15	16	17+	21			
divi	6	13	19	20					
divisor	1	13	20	24	30	43			
int	37								
len	20								
n	28	30	37	41+	42	43			
nb	29	31	32						
newdiv	21	22	23	24					
next	9	11							
prime	22	28							
primes	2	13	20	24					
print	42	44							
raw_input	37								
self	4	5	6	8	9	11	13		
	14	15	17	19	20	25	26		