CATAM Part II - 15.10 - The Continued Fraction Method for Factorization

Introduction

Question 1

We implement the B smoothness test as $B\_smooth(B, N)$, which either returns a list of divisors, or False. To estimate the probability a d-digit integer, ie an integer in range $[10^{d-1}, 10^d - 1]$, is B-smooth with the given set of primes 50, we test all integers up to 10^6 explicitly, then take a random sample of size $900000 = |[10^{6-1}, 10^6 - 1]|$ for higher d. The probabilities we get are:

k	1	2	3	4	5	6	7	8	9	10
estimated probability	1	.888	.488	.215	.0797	.0258	.00743	.00205	.00049	.00011

Table 1: Estimated probability a d digit number is B-smooth

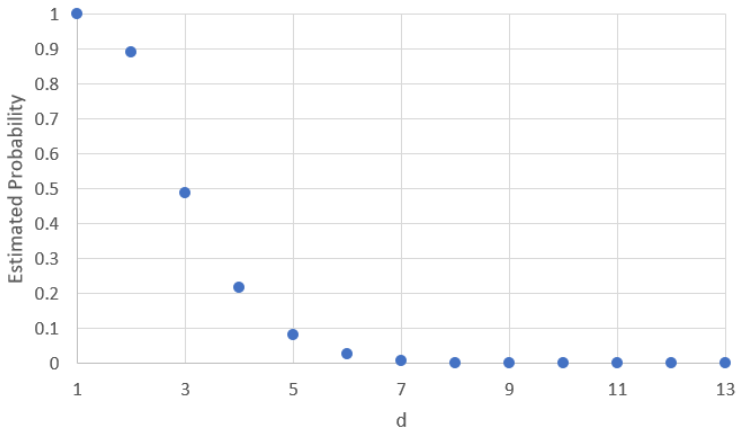

Figure 1: Graph form of results of Table 1

The function written checks divisibility of N by p by calculating $N \mod p$. For very large N we can optimise this by iterating over the digits of N. If $N = a_o a_1 a_2 \dots a_n$ we can initialise $result = 0$ and iterate $result = result \times 10 + a_k \pmod p$ for $k = 0, 1 \dots n$.

Question 2

Lemma. *If $x = \sqrt{N}$ for some positive integer N then each x_n may be written in the form $(r + \sqrt{N})/s$ with r, s integers and $s \mid (r^2 - N)$.*

Proof. Proceed by induction. For the base case $n = 0$ we have $x_0 = c = x = \sqrt{N}$ so $r = 0$, $s = 1$ and so $s \mid (r^2 - N)$. For the inductive step, assume x_n may be written as $(r + \sqrt{N})/s$ with r, s integers satisfying $s \mid (r^2 - N)$. Then compute x_{n+1}.

$$\begin{aligned} x_{n+1} &= \frac{1}{x_n - a_n} \\ &= \frac{1}{\frac{r+\sqrt{N}}{s} - a_n} \\ &= \frac{s}{\sqrt{N} + (r - a_n s)} \\ &= \frac{s(\sqrt{N} - r + a_n s)}{N - r^2 - a_n^2 s^2 + 2r a_n s} \\ &= \frac{\sqrt{N} + (a_n s - r)}{\frac{N-r^2}{s} - a_n^2 s + 2r a_n} \end{aligned}$$

so we have

$$
\begin{aligned}
r' &= a_n s - r \\
s' &= \frac{N - r^2}{s} - a_n^2 s + 2ra_n
\end{aligned}
$$

and thus

$$
\begin{aligned}
r'^2 - N &= a_n^2 s^2 + r^2 - 2a_n sr - N \\
&= s(\frac{N - r^2}{s} - a_n^2 s + 2ra_n) \\
&= ss'
\end{aligned}
$$

ie $s' \mid (r'^2 - N)$ as required.

$\square$

The expressions for r' and s' allow us to store x_n precisely and avoid rounding errors. From IIC Number Theory, we know the sequence of partial quotients is eventually periodic, which can be identified from when x_n repeats. The partial quotients for $\sqrt{N}$, N up to 50 are as follows, with the integers after the colon being repeated:

```
0 ) 0
1 ) 1
2 ) 1 : 2
3 ) 1 : 1, 2
4 ) 2
5 ) 2 : 4
6 ) 2 : 2, 4
7 ) 2 : 1, 1, 1, 4
8 ) 2 : 1, 4
9 ) 3
10) 3 : 6
11) 3 : 3, 6
12) 3 : 2, 6
13) 3 : 1, 1, 1, 1, 6
14) 3 : 1, 2, 1, 6
15) 3 : 1, 6
16) 4
17) 4 : 8
18) 4 : 4, 8
19) 4 : 2, 1, 3, 1, 2, 8
20) 4 : 2, 8
21) 4 : 1, 1, 2, 1, 1, 8
22) 4 : 1, 2, 4, 2, 1, 8
23) 4 : 1, 3, 1, 8
24) 4 : 1, 8
25) 5
26) 5 : 10
27) 5 : 5, 10
28) 5 : 3, 2, 3, 10
29) 5 : 2, 1, 1, 2, 10
30) 5 : 2, 10
31) 5 : 1, 1, 3, 5, 3, 1, 1, 10
32) 5 : 1, 1, 1, 10
33) 5 : 1, 2, 1, 10
34) 5 : 1, 4, 1, 10
35) 5 : 1, 10
36) 6
37) 6 : 12
```

```
38) 6 : 6, 12
39) 6 : 4, 12
40) 6 : 3, 12
41) 6 : 2, 2, 12
42) 6 : 2, 12
43) 6 : 1, 1, 3, 1, 5, 1, 3, 1, 1, 12
44) 6 : 1, 1, 1, 2, 1, 1, 1, 12
45) 6 : 1, 2, 2, 2, 1, 12
46) 6 : 1, 3, 1, 1, 2, 6, 2, 1, 1, 3, 1, 12
47) 6 : 1, 5, 1, 12
48) 6 : 1, 12
49) 7
50) 7 : 14
```

We observe they all have fairly short periods, surprising since the theorem on periodicity doesn't give an indication for what the period will be. We also see that the last partial quotient before repeating is $2\left\lfloor\sqrt{N}\right\rfloor$ TODO

Question 3

The function $q3()$, using $convergents()$ produces the following data. I've included the length of the period for later reference.

```
 3 [2]) −2, 1, −2, 1, −2, 1, −2, 1, −2, 1
 5 [1]) −1, 1, −1, 1, −1, 1, −1, 1, −1, 1
 7 [4]) −3, 2, −3, 1, −3, 2, −3, 1, −3, 2
12 [2]) −3, 1, −3, 1, −3, 1, −3, 1, −3, 1
31 [8]) −6, 5, −3, 2, −3, 5, −6, 1, −6, 5
```

We see the convergents always seem to contain a solution to Pells (positive) equation. Note for N a square, solving either equation gives a solution to $x^2 - y^2 = \pm 1$ which is clearly unsoluble. These results agree with the following theorem, proved in IIC Number Theory.

Theorem. *Let n be the period of the continued fraction expansion of $\sqrt{N}$ for N not a perfect square. Then the convergent (p_{kn-1}, q_{kn-1}) for integer k is a solution to the positive Pell equation for kn even and for the negative Pell equation for kn odd. In particular there are always infinitely many solutions to the positive Pell equation.*

Consider the negative Pell equation $X^2 - NY^2 = -1$. Suppose N is divisible by 4, then mod 4 we get $X^2 \equiv -1 \pmod 4$ which is not soluble so there are no solutions. We can get another condition similarly. Suppose $p \mid N$ with $p \equiv 3 \pmod 4$. Then we get $X^2 \equiv -1 \pmod p$ but $(\frac{-1}{p}) = -1$. So N cannot be divisible by 4 or be congruent to 3 mod 4.

Given x, y, N, to confirm one of Pell's equations holds, we'll use the Chinese Remainder Theorem. Take p a prime, then if $x^2 - Ny^2 = \pm 1$ the same must hold (mod p). By the CRT, the system

$$\begin{aligned} z &= 1 \mod p_1 \\ z &= 1 \mod p_2 \\ &\vdots \\ z &= 1 \mod p_n \end{aligned}$$

has a unique solution mod $p_1 p_2 \dots p_n$. (The same holds with all 1's replaced by -1's). So to check $z = x^2 - Ny^2$ is 1 it suffices the above system holds, as long as the product $p_1 p_2 \dots p_n$ is greater than x^2 and Ny^2. Lets use small primes, up to 113 will do, since the largest Ny^2 can be is 10^{45}.

The reason we are using primes at all is that multiplication can be done without risk of overflow modulo. Consider multiplying x and y mod N. We write for y even, $xy = (2x)(\frac{y}{2})$ and for y odd $xy = x + (2x)(\frac{y-1}{2})$. In the y odd case, add x mod N to the result. Now iterate the process with $x' = 2x$ and $y' = \frac{y}{2}$ or $\frac{y-1}{2}$ accordingly. The process terminates when y becomes 0, which must occur as y becomes strictly smaller each iteration. In doing so we take mod N after every step and are just doing addition so worst case we store at most $2N < 2 * 10^{15}$.

The algorithm is implemented as $modular\_multiply(x, y, mod)$

Our implementation of this method is $verify\_large\_pell(x, y, N)$, which is fairly fast, when tested on x=158070671986249, y=15140424455100, n=109, and x=1766319049, y=226153980, n=61 [1]

We now wish to find some solutions to Pell's Equation. The question doesn't seem to want us to use the stated Theorem, which directly gives a valid convergent. We'll employ a trial and error approach instead, in which we calculate and then test using $verify\_large\_pell(x, y, N)$. We get the following:

```
1) has no solutions (square)
2) (3,2)
3) (2,1)
4) has no solutions (square)
5) (9,4)
6) (5,2)
7) (8,3)
8) (3,1)
9) has no solutions (square)
10) (19,6)
11) (10,3)
12) (7,2)
13) (649,180)
14) (15,4)
15) (4,1)
16) has no solutions (square)
17) (33,8)
18) (17,4)
19) (170,39)
20) (9,2)
21) (55,12)
22) (197,42)
23) (24,5)
24) (5,1)
25) has no solutions (square)
26) (51,10)
27) (26,5)
28) (127,24)
29) (9801,1820)
30) (11,2)
31) (1520,273)
32) (17,3)
33) (23,4)
34) (35,6)
35) (6,1)
36) has no solutions (square)
37) (73,12)
38) (37,6)
39) (25,4)
40) (19,3)
41) (2049,320)
42) (13,2)
43) (3482,531)
44) (199,30)
45) (161,24)
46) (24335,3588)
47) (48,7)
48) (7,1)
49) has no solutions (square)
50) (99,14)
```

[1] Obtained from `https://en.wikipedia.org/wiki/Pell%27s_equation`

```
51) (50,7)
52) (649,90)
53) (66249,9100)
54) (485,66)
55) (89,12)
56) (15,2)
57) (151,20)
58) (19603,2574)
59) (530,69)
60) (31,4)
61) (1766319049,226153980)
62) (63,8)
63) (8,1)
64) has no solutions (square)
65) (129,16)
66) (65,8)
67) (48842,5967)
68) (33,4)
69) (7775,936)
70) (251,30)
71) (3480,413)
72) (17,2)
73) (2281249,267000)
74) (3699,430)
75) (26,3)
76) (57799,6630)
77) (351,40)
78) (53,6)
79) (80,9)
80) (9,1)
81) has no solutions (square)
82) (163,18)
83) (82,9)
84) (55,6)
85) (285769,30996)
86) (10405,1122)
87) (28,3)
88) (197,21)
89) (500001,53000)
90) (19,2)
91) (1574,165)
92) (1151,120)
93) (12151,1260)
94) (2143295,221064)
95) (39,4)
96) (49,5)
97) (62809633,6377352)
98) (99,10)
99) (10,1)
100) has no solutions (square)
500) (930249,41602)
501) (11242731902975,502288218432)
502) (3832352837,171046278)
503) (24648,1099)
504) (449,20)
505) (809,36)
506) (45,2)
507) (1351,60)
508) (44757606858751,1985797689600)
509) (313201220822405001,13882400040814700)
510) (271,12)
```

```
511) (4188548960,185290497)
512) (665857,29427)
513) (13771351,608020)
514) (4625,204)
515) (17406,767)
516) (16855,742)
517) (590968985399,25990786260)
518) (2367,104)
519) (14851876,651925)
520) (6499,285)
521) (32961431500035201,1444066532654320)
522) (19603,858)
523) (81810300626,3577314675)
524) (225144199,9835470)
525) (6049,264)
526) (840560915469529337 75,366501975732429553 2)
527) (528,23)
528) (23,1)
529) has no solutions (square)
530) (1059,46)
531) (530,23)
532) (2588599,112230)
533) (74859849,3242540)
534) (3678725,159194)
535) (1618804,69987)
536) (145925,6303)
537) (192349463,8300492)
538) (9536081203,411129654)
539) (3970,171)
540) (119071,5124)
541)
   (3707453360023867028800645599667005001,1593958697212701100771871387751969 00)
542) (4293183,184408)
543) (669337,28724)
544) (2449,105)
545) (1961,84)
546) (701,30)
547) (160177601264642,6848699678673)
548) (6083073,259856)
549) (1766319049,75384660)
550) (30580901,1303974)
```

It turns out our program was capable of dealing with integers with far more than 15 digits, since python can handle very large **integer** calculations. Even $N = 541$, with smallest x, y of magnitude 10^{36} was able to be solved. Once can verify these are correct via comparing to an online resource, such as [2]

Question 4

The following lemma tells us what we want

Lemma. *Suppose* $x^2 \equiv y^2 \pmod N$ *with* $x \not\equiv \pm y \pmod N$. *Then the GCDs* $(N, x+y)$ *and* $(N, x-y)$ *are both proper divisors of* N.

Proof. Treat $x + y$ and $x - y$ separately, though the arguments are the same
We have $(N, x + y) \mid N$ by definition of GCD. To show it's proper, if $(N, x + y) = N$ then $x \equiv -y \pmod N$, a contradiction. If $(N, x + y) = 1$, considering $x^2 - y^2 \equiv 0 \Rightarrow (x - y)(x + y) \equiv 0 \pmod N$ gives $x - y \equiv 0 \pmod N$, a contradiction.
For the case of $x - y$, the argument is same, replacing y with $-y$. $\square$

The complexity of computing each factor is TODO

[2] http://www.martin-flatin.org/math/pell/pell_equation_1000.xhtml

This method isn't guaranteed to work: For N=6, the squares of 0,1,2,3 are 0,1,4,3 so we can't find x, y as in the Lemma. Similarly for N=10, the squares of 0,1,2,3,4,5 are 0,1,4,9,6,5 so the method won't work for the same reason.

Question 5

We've already discussed how to multiply modulo N in question 3, our implementation being $modular_multiply(x, y, N)$. The function $q5(N, k)$ performs the required task, with k being the largest n such that $P_n.P_n^2$ calculated. On the given N we get the following, with $k = 10$

```
N = 1449774329

P_n: 38075, 38076, 380759, 1561112, 3502983, 8567078, 12070061, 286178481,
    584427023, 870605504, 5258198

P_n^2: 1449705625, 7447, 1449757510, 29495, 1449751962, 52459, 1449771169,
    29137, 1449740329, 37991, 1449753174
```

N=3333999913

```
P_n: 57740, 57741, 230963, 288704, 18419315, 18708019, 55835353, 465390843,
    521226196, 2029069431, 2550295627

P_n^2: 3333907600, 23168, 3333908674, 1791, 3333922345, 37273, 3333987265,
    83849, 3333975752, 83408, 3333986530
```

N=7686335197

```
P_n: 87671, 87672, 263015, 350687, 6926068, 48833163, 153425557, 509109834,
    5703946044, 6213055878, 4230666725

P_n^2: 7686204241, 44387, 7686208649, 8817, 7686311344, 50516, 7686282946, 6503,
     7686221950, 59988, 7686222176
```

Question 6

This question is quite a large part of the IB Core Project 1.1: Matrices over Finite Fields from last year. Since I've switched programming language, I've had to rewrite the code, but have used the same algorithm to perform Gaussian elimination and find an element of the kernel, simplified a bit to work mod 2.

Question 7

The continued fraction algorithm uses elements of the majority of the above questions.
First, define a B-number to be a positive integer x such that all prime factors of $\langle x^2 \rangle$ lie in B, where for $a \in \mathbb{Z}$, $\langle a \rangle$ is the unique integer in $(-\frac{N}{2}, \frac{N}{2}]$ with $\langle a \rangle \equiv a \pmod N$. The algorithm is as follows:

1) Pick a factor base B, which in our case will contain primes ≤ 50, and possibly -1
2) Generate some B numbers $x_1, x_2, \ldots, x_k$. It turns out P_n have a good chance of being B-numbers by LEMMA SOMEHTING
3) Find a non-empty subset $I \subset \{1, \ldots, k\}$ such that $\prod_{i \in I} \langle x_i^2 \rangle = y^2$ is a square.
4) For $x = \prod_{i \in I} \langle x_i \rangle$ satisfies $x^2 \equiv y^2 \pmod N$. So by the earlier lemma, we can find some non trivial factors of N if $x \not\equiv y \pmod N$.

Note we've changed the q5() function slightly to give us $\langle P_n^2 \rangle$ instead of $P_n^2 \pmod N$
EXPLAIN ALGORITHM
The output for various N is given below

```
N = 9509

The algorithm uses P_n for n = 0,2,3
The corresponding B numbers are 97,195,3413
Multiplying the B numbers, x_i gives an x of 294
```

```
Multipling <x_i^2> gives a y equal to 9289 = (−1) x 2^2 x 5 x 11
This gives us factors gcd(x+y,N) = 37and gcd(x−y,N) = 257
```

N = 14429

```
The algorithm uses P_n for n = 0,2
The corresponding B numbers are 120,3003
Multiplying the B numbers, x_i, gives an x of 14064
Multipling <x_i^2> gives a y equal to 14371 = (−1) x 2 x 29
This gives us factors gcd(x+y,N) = 47 and gcd(x−y,N) = 307
```

N=1449774329

```
The algorithm uses P_n for n = 43
The corresponding B numbers are 1245500098
Multiplying the B numbers, x_i, gives an x of 1245500098
Multipling <x_i^2> gives a y equal to 145 = 5 x 29
This gives us factors gcd(x+y,N) = 51043 and gcd(x−y,N) = 28403
```

N=3333999913

```
The algorithm uses P_n for n = 6,22,45,100,168
The corresponding B numbers are
   55835353,466038032,1372728391,2510428257,1696557395
Multiplying the B numbers, x_i, gives an x of 2938205297
Multipling <x_i^2> gives a y equal to 2220030420 = (−1)^2 x 2^6 x 3^4 x 13 x 17
   x 29 x 31 x 41
This gives us factors gcd(x+y,N) = 99991 and gcd(x−y,N) = 33343
```

N+7686335197

```
The algorithm uses P_n for n = 15,130,152
The corresponding B numbers are 2002379263,1821227876,6615421364
Multiplying the B numbers, x_i, gives an x of 7393655649
Multipling <x_i^2> gives a y equal to 7668282421 = (−1) x 2^3 x 3^2 x 7^3 x 17 x
    43
This gives us factors gcd(x+y,N) = 93257 and gcd(x−y,N) = 82421
```