MINTY数の探究

―What Makes Us Excited Is THE MINTY NUMBER

秋田県立横手高等学校　２年１組　数学班

安田実生　池田晴香　武田侑純　成田皓城　三浦央暉

MINTY数の誕生

私たちの学年主任F先生の車のナンバーがラマヌジャンのタクシー数の“1729”であったことから、タクシー数に興味をもった。しかし、タクシー数は立方数の和であるため扱いにくいことから、平方数の和に変えて研究を進めることにした。

また、名前の由来は三浦のM、池田のI、成田のN、武田のT、安田のYである。

【定義】

２つの平方数の和としてn通りに表される最小の正の整数。

$$M(n) = x^2 + y^2 = a^2 + b^2 = m^2 + n^2 = \ldots$$

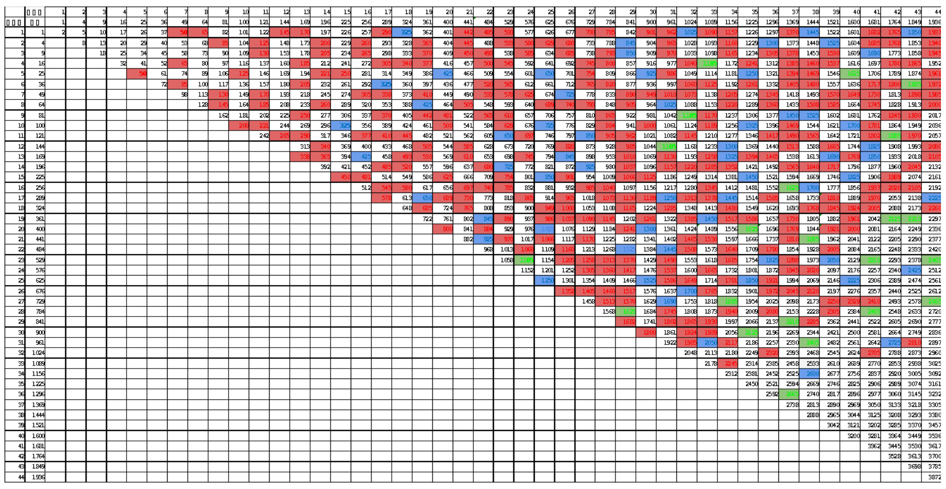
仮説

MINTY数には規則性があり、漸化式に表すことができる。

検証１

検証方法①

Googleのスプレッドシートを用いて、縦の列と横の行に1以上40000以下の平方数同士の和を1^2+1^2から200^2+200^2のすべての組み合わせを表示させて重複する数値に色をつける。



【結果】

M(1)=2　M(2)=50　M(3)=325　M(4)=1105　M(5)=8125　M(6)=5525

M(7)なしM(8)=27625

【考察】

- ●MINTY数には単純増加の規則性が見られない。
- ●M(6)はM(5)よりも小さくなり、nの変化とM(n)の変化の仕方には関係性がない。

※もとになったタクシー数にも規則性はないと言われている。

検証方法①の作業効率がかなり悪い
⇒プログラミングに挑戦しよう！！

検証方法②

プログラミング言語はpythonを使用。

１から30000000までの数の中にMINTY数がどのくらいあるのかを調べるプログラムと、見つけたMINTY数はどのような数の二乗の組み合わせでできているのかを調べるプログラムを作成し、それらを使い調べる。

↓MINTY数を見つける

```
import math
import time

n=25000000

minty = {0:0}
start = time.time()

for r in range(1,n):

    rad = int(math.sqrt(r/2))
    count =0

    for x in range(1,rad+1):
        ytem = math.sqrt(r-x**2)
        if int(ytem) == ytem :
            count += 1

    if r % 1000000 == 0 :
        print('****** {0} ******'.format(r))
    if not count in minty.keys():
        print('{1:2}番目のMinty数を発見！　ーーー>　{0:8}'.format(r,count))
        minty[count]=1
    else:
        minty[count]+=1

print('¥n')

for u,v in sorted(minty.items()):
    print('{0:2}個の交点は{1:8}パターン'.format(u,v))
elapsed_time = time.time() - start
print ("elapsed_time:{0}[sec]".format(elapsed_time))
```

↓平方数の和をつくる組み合わせを見つける

```
import collections

minty={}
num=500

for i in range(1,num):
  for j in range(i,num):
    com=i**2+j**2
    minty[(i,j)]=com

count=0
for c in collections.Counter(minty.values()).most_common():
  if c[0]==5525:
    keys=[k for k,v in minty.items()if v == c[0]]
    print('({0:9},{1:3})-'.format(c[0],c[1])+str(keys))
```

```
(      5525,  6)-[(7, 74), (14, 73), (22, 71), (25, 70), (41, 62), (50, 55)]
```

【仮説】

$M(6)$と$M(5)$のように、nの変化のしかたと$M(n)$の変化のしかたが一致しない。

【結果】

M(7)=105625　M(9)=71825　M(10)=138125　M(11)=5281250

M(12)=160225　M(13)=12211025　M(14)=2442050　M(15)=1795625

M(16)=801125　M(18)=2082925　M(20)=4005625　M(24)=5928325

M(32)=29641625

※M(17),M(19),M(21)~M(23)とM(32)を除くM(25)以降にはない

【考察】

- ●MINTY数の(1)=2以外はすべて5の倍数
  - ⇒MINTY数は4で割ると、必ず余りが1または2になる
  - ※これに関して証明することはまだできていないが、大きな数になってもこの性質が変わらないことから、MINTY数がこの性質をもつことを十分に期待できる。
- ●やはりMINTY数に規則性はない
  - ⇒nが素数のときだけMINTY数が急激に大きくなっている

検証２

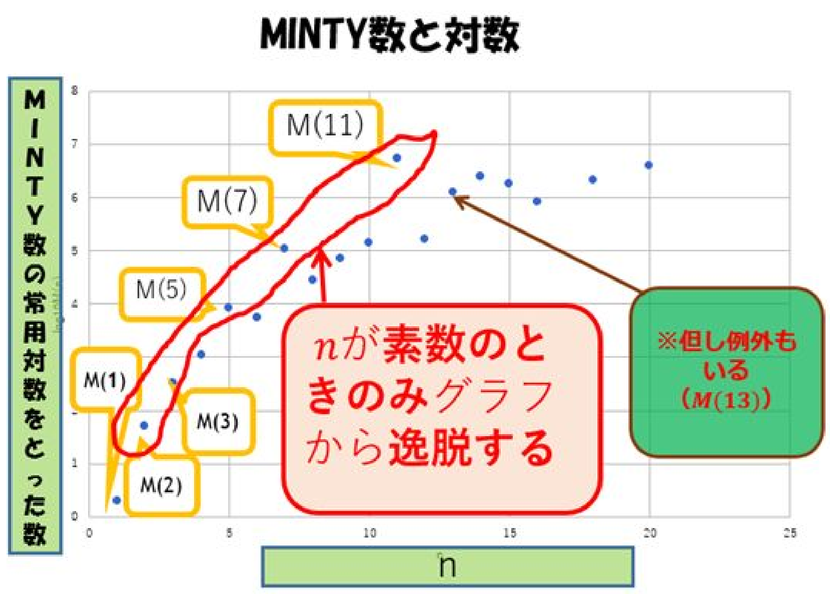
検証方法②

10を底とする対数、すなわち常用対数を用いて扱いやすい桁数にまで下げる処理をし、MINTY数のグラフを作成する。

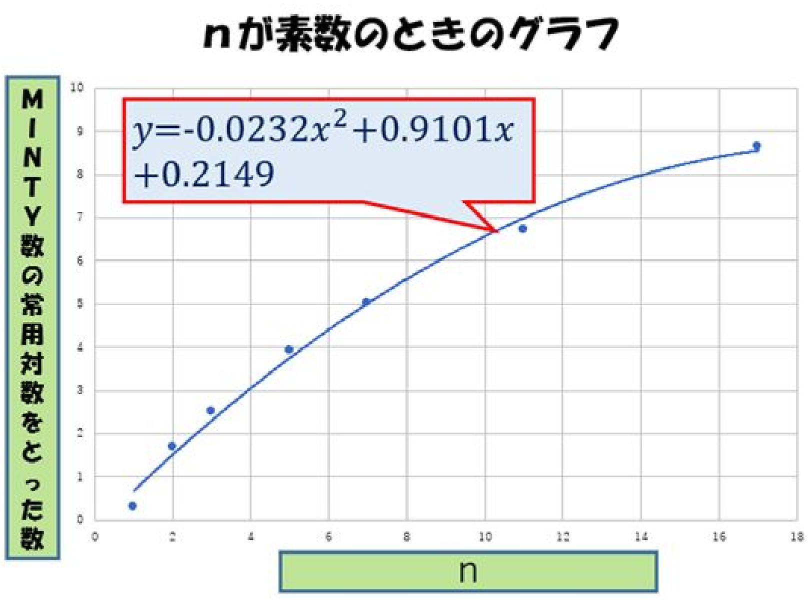
求めたのMINTY数全体のグラフが下図（ｉ）のようになる。横軸をnの値、縦軸をMINTY数の常用対数とし、散布図を作成した。図（ｉ）より、グラフは二次関数のような増え方をしているのが分かる。そして、この赤枠で囲まれた点から、nが素数のときだけ、グラフのおおよその線から外れていることが読み取れる。ただしM(13)だけは例外のようで、素数の動きからも外れている。

次に先ほどの例外だったM(13)を除いた、赤枠で囲んだnが素数のときのMINTY数のデータだけを抽出して作成したグラフが下図（ⅱ）のようになる。このグラフも前のグラフと同じ縦軸と横軸をとった。このグラフ上の点を大体通るように引いた曲線、すなわち近似曲線を描いたところ、最も自然なグラフとして、二次関数のグラフとなった。

ｉ）



ⅱ）



【考察】

●思いのほかきれいに曲線が引けたので、この増加の仕方を保つならば、ｎが素数のときのMINTY数はこの近似直線のライン上に近い位置にあるかもしれない。しかし、素数に関する謎自体も多く、素数そのものに規則性がないことや、M(13)のような例外も存在することから、今のところははっきりと断定することはできないという結論になった。