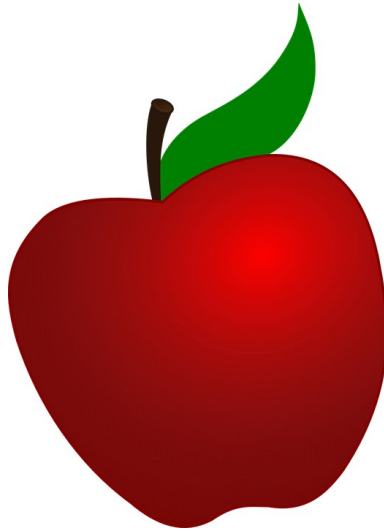# K-tuples

all about sets of prime numbers project

Calculations and coding and computer searches and writeup

by Matthew Charles Anderson

January 2022

email matthewcharlesanderson2@gmail.com

< joke > For those of you unfamiliar with prime numbers, these are prime k-tuples, not Keurig k-cups. This is prime numbers, not coffee. < end joke >

First part
Introduction

Hopefully, you will find this note book interesting. I sure did.

This note is all about sets of prime numbers.

Sets of prime numbers are also known as k-tuples

Let k be a positive integer. That is k is a counting number.   So, k is in the set 1,2,3, ...

See that k is a member of {1,2,3, …}

These k-tuples are similar to prime constellations. A prime constellation is a k-tuple with added restrictions. Essentially, a prime constellation is a k-tuple with consecutive primes, and they are as close together as possible. Prime constellations will necessarily be consecutive primes. K-tuples have the freedom for more distance between the primes.

The search for prime numbers is fun for me. In the modern world, this involves the use of a computer. Computers can do calculations, and search and find special prime numbers. Many people have recorded the largest prime number, or sets of prime numbers that they could find. See for example mersenne.org for the world's largeset prime number, which is a Mersenne number. Mersenne numbers have the form $M(p) = 2^p - 1$.. I have concentrated on lists of the smallest prime numbers with a certain pattern. I concentrate on the difference between the primes.

Two places to find lists of prime numbers are OEIS.org and pzKtupel.de. The Online Encyclopedia of Integer Sequences (OEIS) was started in the 1960s and has over 300,000 sequences of numbers. I have contributed to this database several times. A list of simple prime numbers can be found at oeis.org/A40. Or see oeis.org/A022009 for a 7-tuple that I authored. I used Maple code to produce lists of prime sets (7-tuples through 13-tuples). I did most of this around the year 2012. I did original calculations. That is, I had my computer produce numbers that had never been printed out before. I think that is kind of cool.

See web page – https://mattanderson.fun/f/prime-constellations

This web page is paid for by me, and rented from an internet service provider (ISP).

Also, thanks to Norman Luhn and his webpage http://www.pztupel.de/ktuplets.htm . Ktuplets are a very similar topic to k-tuples and prime constellations. They also involve prime numbers.

Although most prime numbers are still a mathematical curiosity, I do it for the sport of it. I consider this interesting mathematical trivia. But who knows, maybe in 1,000 years some researcher will look it up and have a use for the data. He will be happy that I, and others, did this. Currently, some mathematics is used in credit card transactions. Also, some prime numbers have a patent number. An overnight computer run is not uncommon in the search for certain prime numbers. In fact, some people keep their computers running all the time. This way the computers are actively doing something constantly. For me, it is fun to tell a computer what to do.

2ⁿᵈ part
computer code

Maple is a high level computer algebra system. It is easy for me to write Maple code. I use the isprime() command to determine prime-ness. The C language also has a library for large integers. There are GNU and GMP libraries for C. C might be faster than Maple for program execution. Here is my Maple code for prime pairs. It has one passed variable, for the difference between the two prime numbers.

```
# This is Maple code
MattsPrimePairsProc := proc(diff1)
local a, counter, searchstop;
counter := 1;
searchstop := 1000;
for a from 3 by 2 to searchstop do          # just search odd numbers for primes.  2 is only even prime
if isprime(a) and isprime(a+diff1)
then print(counter, "Hourah!! we found a pair",a, " ",a+diff1);  counter := counter + 1
end if;
end do;
end proc;
# end code block
```

That was the code for my pairs procedure. Later, I make searchstop a passed parameter.

This bit of computer code is written in the Maple language. Maple is a computer algebra system. Maple was developed in Waterloo, Canada.

Here is the computer code, using the Maple language.

```
<begin code>
MattsPrimePairsProc := proc(searchstop,diff1)
# there are two passed parameters.
# searchstop is to determine list length
# diff1 determines the difference between the two prime numbers.

local a, counter;
counter :=1;
print("This is a search for pairs of prime numbers.  The count and smaller number are displayed.");
for a from 3 to searchstop by 2 do
if isprime(a) and isprime(a+diff1)
then print(counter, a);
counter := counter +1;
end if;
end do;
end proc;
<end code>
```

Enjoy the rest of the data sets.
Here is the output from this pairs procedure.

#Lets find prime pairs with difference of 46
note OEIS.org only records prime pairs for even numbers 2 through 44.

test run
MattsPrimePairsProc(50,46);   #50 is max prime and 46 is difference between primes

1, 7
2, 13
3, 37
4 43
and that is our data set for input (50,46).

# test run number 2
MattsPrimePairsProc(20,48)
1,5
2,11
3,13
4,19
seems to work just fine
here we have found prime pairs with a difference of 48 and the smaller of the pair is less than 20.

# test run number 3
MattsPrimePairsProc(50,50)
1,3,53
2,11,61
3,17,67
4,23,73
5,29,79
6,47,97
That was a nice data set.

# test run number 4
MattsPrimePairsProc(30,54)
1,5,59
2,7,61
3,13,67
4,17,71
5,19,73
6,29,83
more interesting data

MattsPrimePairsProc(80,100)
1,3,103
2,7,107
3,13,113

```
4,31,131
5,37,137
6,67,167
7,73,173
8,79,179
9,97,197
# good data

# now a data set with difference of 1,000.
MattsPrimePairsProc(100,1000)
1,13,1013
2,19,1019
3,31,1031
4,61,1061
5,97,1097
# more interesting prime number data.  I did not know that 97 and 1097 are prime numbers.
```

more about 2-tuples

These 2- tuples are already in Online Encyclopedia of Integer Sequences (OEIS.org)

Prime numbers are interesting to me. We list pairs of prime numbers. The Online Encyclopedia of Integer Sequences has lots of fun sequences.

Below is a list such that both p and p+e are prime and e is an even number.

In mathematics, a conjecture is a statement that is probably true. This is different from at theorem, which can be proved.

In number theory, Polignac's conjecture was made by Alphonse de Polignac in 1849 and states: For any positive even number n, there are infinitely many cases of two consecutive prime numbers with difference n.

Numerical evidence suggests that Polignac's conjecture is true for all positive even numbers n.
As of February 2022, Polignac's conjecture is not a theorem, it is still a conjecture. A generalization is the k-tuple conjecture, which states that there are probably infinitely many prime numbers for all admissible k-tuple patterns. Just to let you know, a pattern is admissible if it has no divisibility problems for any prime number p and p<k. For example, 2 is the only even prime number. So the pattern (0,1) means we want two prime numbers with difference 1. All prime numbers greater than 2 are odd, so {2,3} is the only prime pair with pattern (0,1) and this is not an admissible pattern. Similarly, we can do the same thing with a 3-tuple and a divisibility by 3 argument. Suppose we want odd primes that have remainders 0,1 and 2 when divided by 3. Our only solution is {3,5,7} and (0,2,4) is not an admissible pattern. Because, 3 is the only prime number that is divisible by 3. This makes sense to me. I'm sure it makes sense to some readers.

OEIS (Online Encyclopedia of Integer Sequences) A number, description of prime pair

Enumeration Counts (length of number lists in OEIS) as of January 2021

A001359 p,p+2 enumeration count 100,000 These are known as twin primes.
A023200 p,p+4 enumeration count 10,000 called cousin primes
A023201 p, p+6 enumeration count 10,000 called sexy primes
A023202 p,p+8 enumeration count 10,000 unofficially called octa primes
A023203 p,p+10 enumeration count 10,000 unofficially called deca primes
A046133 p,p+12 enumeration count 1,000
A153417 p,p+14 enumeration count 1,000
A049488 p,p+16 enumeration count 10,000
A153418 p,p+18 enumeration count 1,000
A153419 p,p+20 enumeration count 10,000
A242476 p,p+22 enumeration count 1,000
A033560 p,p+24 enumeration count 1,000
A252089 p,p+26 enumeration count 52
A252090 p,p+28 enumeration count 10,000
A049481 p,p+30 enumeration count 10,000
A049489 p,p+32 enumeration count 10,000
A252091 p,p+34 enumeration count 10,000

A156104 p,p+36 enumeration count 1,000
A271347 p,p+38 enumeration count 10,000
A271981 p,p+40 enumeration count 10,000
A271982 p,p+42 enumeration count 10,000
A272176 p,p+44 enumeration count 10,000
pattern for p,p+46 is not in
The Online Encyclopedia of Integer sequences as of March 2020.

These sequences with difference > 44 are still determined to be
'not of general interest'.   But that could change with time and interest of others.

The next logical step is 3-tuples. And, there are several 3-tuples in OEIS and in other public web pages
and databases.

11-tuples

Here is some more Maple code. The search depth (and thus the list length) is specified with the parameter 'searchstop'. Also, the 11 prime numbers are specified.   The pattern of differences between the 11 prime is passed into the procedure tuple10().

The variable names are self explanitory. 'a' is our loop variable. We ignore the prime number 2. All prime numbers bigger than 2 are odd numbers. So we only need to search the odd numbers.

The Maple computer language has a useful function called isprime(). It returns true or false. If isprime fails for a number, then that number > 10^80. We will not be dealing with numbers with more than 28 digits.

```
#Maple code
tuple10 := proc(searchstop, diff1, diff2, diff3, diff4, diff5, diff6, diff7, diff8, diff9, diff10);
      local a, counter;
      counter := 1;                    # We count the number of prime number sets that are found.
      Print(" Prime set search procedure");
      print(" search depth",searchstop");
      print(" pattern", diff1,diff2, diff3, diff4, diff5, diff6, diff7, diff8, diff9, diff10);
      print(" note that if a pattern variable is 0, then the lead prime is repeated.");
      For a from 3 to searchstop by 2 do
              if isprime(a) and isprime(a+diff1) and ispirme(a+diff2) and isprime(a+diff3) and
              isprime(a+diff4) and isprime(a+diff5) and isprime(a+diff6) and isprime(a+diff7) and
              isprime(a+diff8) and isprime(a+diff9) and isprime(a+diff10)
              then
                      print(counter, a, a+diff1, a+diff2, a+diff3, a+diff4, a+diff5, a+diff6, a+diff7,
                      a+diff8, a+diff9, a+diff10);
                      counter := counter+1;
              end if;
      end do;
end proc;
# end code block
```

Here is some code output.

Tuple10(30,12,0,0,0,0,0,0,0,0,0,0);
1,5,17,5,5,5,5,5,5,5,5,5
2,7,19,7,7,7,7,7,7,7,7,7
3,11,23,11,11,11,11,11,11,11,11,11
4,17,29,17,17,17,17,17,17,17,17,17

# now for a 3-tuple with pattern (0,12,14). And 90 is search depth.
Tuple10(60,12,14,0,0,0,0,0,0,0,0,0);
# I'm tired of typing all those repeated prime numbers. So I shorten the output
1, 5, 17, 19
2, 17, 29, 31
3, 29, 41, 43
4, 47, 59, 61
5, 59, 71, 73

now for an admissible 11-tuple.
#Lists of k-tuples with non-admissible patterns are finite.
It is like asking for even and odd prime numbers. 2 is the only even prime.
There are infinitely many odd prime numbers.
So the pattern (0,1) yealds the finite list of prime numbers {2,3}.
If the k-tuple conjecture is true, then all admissible k-tuples are potentially infinite in length.

Hope that explains 'admissible' and 'k-tuple conjecture.

# code output for admissible 10-tuple pattern
tuple10(12,0,2,6,8,14,20,26,32,38,44);
1,3,5,9,11,17,23,29,35,41,47
2,5,7,11,13,19,25,31,37,43,49
3,11,13,17,19,25,31,37,43,49,55

That's all for now.