Lecture 3:
Introduction to Computing and PARI

William Stein

Math 124 HARVARD UNIVERSITY **Fall 2001**

1 Introduction

> "The object of numerical computation is theoretical advance." – *Bryan Birch describing A. O. L. Atkin.*

Much progress in number theory has been driven by attempts to prove conjectures. It's reasonably easy to play around with integers, see a pattern, and make a conjecture. Frequently proving the conjecture is *extremely difficult.* In this direction, computers help us to

- find more conjectures
- disprove conjectures
- increase our confidence in a conjecture

They also frequently help to solve a specific problem. For example, the following problem would be hopelessly tedious by hand. Here's an example of such a problem:

> Find all integer $n < 100$ that are the area of a right triangle with integer side lengths.[1]

This problem can be solved by a combination of very deep theorems, a few big computer computations, and a little luck.

2 Some Assertions About Primes

A computer can quickly "convince" you that many assertions about prime numbers are true. Here are three.

- *The polynomial $x^2 + 1$ takes on infinitely many prime values.*
  Let
  $$f(n) = \{x : x < n : x \text{ and } x^2 + 1 \text{ is prime }\}.$$
  With a computer, we quickly find that
  $$f(10^2) = 19, \quad f(10^3) = 112, \quad f(10^4) = 841, \quad f(10^5) = 6656.$$
  Surely $f(n)$ is unbounded! The PARI code to compute $f(n)$ is very simple:

[1]We will discuss the "The Congruent Number Problem" in more depth later in this course.

```
? f(n) = s=0; for(x=1,n,if(isprime(x^2+1),s++)); s
? f(100)
%1 = 19
? f(1000)
%2 = 112
? f(10000)
%3 = 841
? f(100000)
%4 = 6656
```

- *Every even integer $n > 2$ is a sum of two primes.*
  With a computer we find that this seems true

n	p	q
4	2	2
6	3	3
8	3	5
10	3	7
12	5	7

... and much further. In practice, it's easy to write an even number as a sum of two primes. Why should there be any weird even numbers out there for which this can't be done? PARI code to find p and q:

```
? gb(n) = forprime(p=2,n,if(isprime(n-p),return([p,n-p])));
? gb(4)
%7 = [2, 2]
? gb(6)
%8 = [3, 3]
? gb(100)
%9 = [3, 97]
? gb(1000)
%10 = [3, 997]
? gb(570)          \\ takes no time at all!
%11 = [7, 563]
```

- *There are infinitely many primes p such that $p + 2$ is also prime.*
  Let $t(n) = \#\{p : p \leq n \text{ and } p + 2 \text{ is prime }\}$. Using a computer we quickly find that

$$t(10^2) = 8, \quad t(10^3) = 35, \quad t(10^4) = 205, \quad t(10^5) = 1024.$$

The PARI code to compute $t(n)$ is very simple:

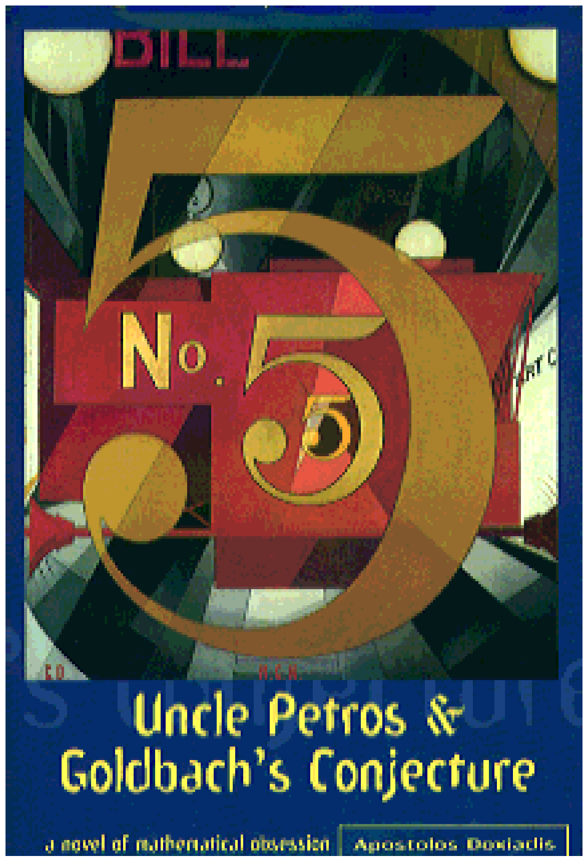
```
? t(n) = s=0; forprime(p=2,n,if(isprime(p+2),s++)); s
? t(10^2)
%12 = 8
? t(10^3)
%13 = 35
```

```
? t(10^4)
%14 = 205
? t(10^5)
%15 = 1224
```

Surely $t(n)$ keeps getting bigger!!

As it turns out, these three assertions are *all* OLD famous extremely difficult unsolved problems! Anyone who proves one of them will be very famous.

Assertion 2 is called "The Goldbach Conjecture"; Goldbach reformulated it in a letter to Euler in 1742. It's featured in the following recent novel:



The publisher of that novel offers a MILLION dollar prize for the solution to the Goldbach conjecture:

`http://www.faber.co.uk/faber/million_dollar.asp?PGE=&ORD=faber&TAG=&CID=`

The Goldbach conjecture is true for all $n < 4 \cdot 10^{14}$, see

`http://www.informatik.uni-giessen.de/staff/richstein/ca/Goldbach.html`

Assertion 3 is the "Twin Primes Conjecture". According to

`http://perso.wanadoo.fr/yves.gallot/primes/chrrcds.html#twin` on May 17, 2001, David Underbakke and Phil Carmody discovered a 32220 digits twin primes record with a set of different programs: $318032361 \cdot 2^{107001} \pm 1$. This is the current "world record".

With a computer, even if you can't solve one of these "Grand Challenge" problems, at least you can perhaps work very hard and prove it for more cases than anybody before you, especially since computers keep getting more powerful. This can be very fun, especially as you search for a more efficient algorithm to extend the computations.

3 Some Tools for Computing

Calculator: A TI-89 can deal with integers with 1000s of digits, factor, and do most basic number theory. I am not aware if anyone has programmed basic "elliptic curve" computations into this calculator, but it could be done.

Mathematica and Maple: Both are commercial, but they are very powerful, can draw pretty pictures, and there are elliptic curve packages available for each (`apecs` for Maple, and something by Silverman for Mathematica).

PARI: Free, open source, excellent for our course, simple, runs on Macs, MS Windows, Linux, etc.

MAGMA: Huge, non-free but nonprofit, what I usually use for my research. I can legally give you a Linux executable if you are registered for 124.

My Wristwatch: Perhaps the only wristwatch in the world that can factor your social security number? `:-)`

4 Getting Started with PARI

4.1 Documentation

The documentation for PARI is available at

`http://modular.fas.harvard.edu/docs/`

Some PARI documentation:

1. **Installation Guide:** Help for setting up PARI on a UNIX computer.
2. **Tutorial:** 42-page tutorial that starts with `2 + 2`.
3. **User's Guide:** 226-page reference manual; describes every function
4. **Reference Card:** hard to print, so I printed it for you (handout)

4.2 A Short Tour

```
$ gp
Appele avec : /usr/local/bin/gp -s 10000000 -p 500000 -emacs

                  GP/PARI CALCULATOR Version 2.1.1 (released)
```

```
                i686 running linux (ix86 kernel) 32-bit version
                (readline v4.2 enabled, extended help available)

                       Copyright (C) 2000 The PARI Group

PARI/GP is free software, covered by the GNU General Public License, and
comes WITHOUT ANY WARRANTY WHATSOEVER.

Type ? for help, \q to quit.
Type ?12 for how to get moral (and possibly technical) support.

   realprecision = 28 significant digits
   seriesprecision = 16 significant terms
   format = g0.28

parisize = 10000000, primelimit = 500000
? \\ this is a comment
? x = 571438063;
? print(x)
571438063
? x^2+17
%2 = 326541459845191986
? factor(x)
%3 =
[7 1]

[81634009 1]
? gcd(x,56)
%5 = 7
? x^20
%6 = 13784255037665854930357784067541250773222915495828020913935
8450113971943932613097560462268162512901194466231159983662241797
60816483100648674388195744425584150472890085928660801
```

4.3 Help in PARI

```
? ?
Help topics:
  0: list of user-defined identifiers (variable, alias, function)
  1: Standard monadic or dyadic OPERATORS
  2: CONVERSIONS and similar elementary functions
  3: TRANSCENDENTAL functions
  4: NUMBER THEORETICAL functions
  5: Functions related to ELLIPTIC CURVES
  6: Functions related to general NUMBER FIELDS
  7: POLYNOMIALS and power series
  8: Vectors, matrices, LINEAR ALGEBRA and sets
```

```
  9: SUMS, products, integrals and similar functions
 10: GRAPHIC functions
 11: PROGRAMMING under GP
 12: The PARI community

Further help (list of relevant functions): ?n (1<=n<=11).
Also:
  ? functionname (short on-line help)
  ?\             (keyboard shortcuts)
  ?.             (member functions)
Extended help looks available:
  ??             (opens the full user's manual in a dvi previewer)
  ??  tutorial   (same with the GP tutorial)
  ??  refcard    (same with the GP reference card)

  ??  keyword    (long help text about "keyword" from the user's manual)
  ??? keyword    (a propos: list of related functions).
? ?4

addprimes     bestappr      bezout        bezoutres     bigomega
binomial      chinese       content       contfrac      contfracpnqn
core          coredisc      dirdiv        direuler      dirmul
divisors      eulerphi      factor        factorback    factorcantor
factorff      factorial     factorint     factormod     ffinit
fibonacci     gcd           hilbert       isfundamental isprime
ispseudoprime issquare      issquarefree  kronecker     lcm
moebius       nextprime     numdiv        omega         precprime
prime         primes        qfbclassno    qfbcompraw    qfbhclassno
qfbnucomp     qfbnupow      qfbpowraw     qfbprimeform  qfbred
quadclassunit quaddisc      quadgen       quadhilbert   quadpoly
quadray       quadregulator quadunit      removeprimes  sigma
sqrtint       znlog         znorder       znprimroot    znstar

? ?gcd
gcd(x,y,{flag=0}): greatest common divisor of x and y. flag is optional, and
can be 0: default, 1: use the modular gcd algorithm (x and y must be
polynomials), 2 use the subresultant algorithm (x and y must be polynomials).

? ??gcd
\\ if set up correctly, brings up the typeset section from the manual on gcd
```

We will discuss writing more complicated PARI programs on October 10.