Boston University
Summer I 2008/2009

MA341 Number Theory
Kalin Kostadinov

Workshop 2- Computers and Number Theory

Mathematics is also about observing patterns in the results of various computations.

The old mathematicians were vey skillful at computing, but even they sometimes were hiring other people to compute vast tables of data, from which the mathematician will then look for patterns.

The advent of computers made data easily available for everyone.

In this workshop, we will learned how to use a special software, named PARI/GP, to make computations in number theory.

Here are the (very-slightly edited, there are inaccuracies reflecting the student level of understanding of some of the concepts in the workshop) comments of some of the students about the activity:

For the second workshop we worked with PARI to learn the basics of working with number theory and computers. We examined several examples and then worked through four problems ourselves.
These ranged from computing the solutions to Diophantine equations to finding twin primes in a given range of numbers. I thought the workshop was useful in showing the basics of how computers can be put to use in the field of number theory.
The most interesting result which we encountered was in the solution to problem 4.
This was the table of remainders which was used in Tuesday's lecture to lead into Fermat's Little Theorem. Incidently this was also the problem which gave me the most trouble.
I thought the workshop tied into the lectures well.
-Andrew
****

In lab 2, we used PARI to implement a few trivial programs: to solve for the first taxicab number, output a list of twin primes between 1,000,000 and 1,001,000 and to find the first 19 consecutive composite numbers.
The new language was neat, and apparently had some pretty quick implementations of isprime().

I also collaborated with Dave and helped him fix some easy to make, hard to discover errors in the twin primes program like using ==! when it is meant to use the operator "!=".

I ended up installing PARI on my computer at home, and used it to solve problem C in homework 5.
It saved me the time of having to write an isprime() function in C, which was nice.

Joe
****

Our second workshop focused on understanding Number Theory with the help of computer programming and language. For the workshop, we used a program called PARI. This workshop was a little challenging for me because although some of my professors rarely used programs such as Matlab

or Mathematica to demonstrate some equations and graphs, I never really used any of those programs my own. However, it wasn’t challenging in a discouraging way but in a way that the more I got used to the program and saw the outputs, the more I enjoyed it. The exercise I most enjoyed out of 4 was writing a program for sum of the primes. The reference sheet was very helpful in terms of introducing new shortcut commands like prime(n) and isprime(x) which were very useful and made the programming a lot less complicated. Overall I really enjoyed Workshop II, and to be honest I enjoyed it a lot more than Workshop I.
Cicek

****

In workshop on Monday we wrote several programs in PARI that solved number theoretic problems through basic algorithms. We wrote programs to solve diophantine equations, to find information about the sequence of primes (i.e. finding twin primes and finding sequences of all composite numbers), and to give a table of a^n mod p for various a, n, and p. Because our algorithms largely relied on brute force (i.e. checkinh every number to see if it is a solution), probably the most important things we learned were the functions of for-loops (and nested for loops for checking solutions to equations in several variabless) and if statements.
Also, the tables of a^n mod p revealed several interesting results, such as a^(p-1) is congruent to 1 (p) and (p-1)^n is congruent to either 1 or (p-1) (because p-1 is congruent to -1). The former result is generalized by Fermat's little theorem and then by Euler phi-theorem.

David

****

In this Workshop we used the mathematics software program PARI to write programs in order to solve different number theory problems
To begin with we looked at a program which took a number as an input from the user and determined whether the number was a complete square. If it was it would print “YES” and if wasn't it would print “NO”.
In the next program we learned how to use the commands FOR which set the boundaries for our variable and IF which we used to set different conditions. We also learned how to start a cycle and BREAK out of the cycle.
In the third example we looked at a program which took a number as an input from the user. If the number was higher than 30 and less than 1 it would say that it was out of range. For any number in the range the program would calculate the factorial of that number.
Next we wrote a program which found the solution to the Diophantine equation x^3+y^3-1728=0.
We set the boundary of x to be between zero and twenty. It would solve the equation and print the results for x and y if they satisfied the Diophantine equation.
In the next example we tried to write a program which found the pairs of twin primes between 1000000 and 1001000. In order to write this program we made use of the command ISPRIME(x) and ISPRIME(x+2) to see if we get a twin primes pair.
Next we wrote a program which calculated the first 19 consecutive non-prime numbers where we made use of the nextprime(x) command.
Next we wrote a program to compute the remainders of consecutive powers of a number modulo a

given name. We saw some interesting patterns in the results of this program. One result was that if a==b(mod m) the a^n==b^n (mod m)
Also the column for a which was a=p-1 had a pattern of (p-1) and then (1) continued
There was also a row of only zeros for each prime for the power which was one less than the prime number. For a's that were equal to the prime number the column was just zeros.

-Tina

****

I really enjoyed doing a bit of programming today. I forgot how fun a little bit of computer science can be.

- Susan

****

Write up for Workshop 6/1/09

1) A=3; P=7; for(e=1,2*P,print(A^e%P))

```
2) when A=3;P=7: 3,2,6,4,5,1,3,2,6,4,5,1,3,2
   when A=3;P=5: 3,4,2,1,3,4,2,1,3,4
   when A=4;P=5: 4,1,4,1,4,1,4,1,4,1
   when A=4;P=7: 4,2,1,4,2,1,4,2,1,4,2,1,4,2
   when A=10;P=11: 10,1,10,1,10,1,10,1,10,1,10,1,10,1,10,1,10,1,10,1,10,1
```

3) After I found the results for my first two examples I made the conjecture that the output starts to repeat after P-1. However that does not hold up and it appears that when P-A=1 the output repeats every two solutions (A,1). I checked with A=6,P=7 and this still holds up. For A=4,P=5 it makes logical sense when you take a glance at the A^e in relation to the multiples of 5 (alternating between 4mod5 and 1mod5). I am curious and hope to keep studying this problem and discover why/prove this for a general A and P.

I am very glad that we did this workshop however it was frustrating at times because I have never been exposed to any form of programming or computer language prior to yesterday. I realized how working with computers forces the human to think in a logical way and then translate into the computer language. I am also curious to test the power of the computer and ask questions that the human mathematician may struggle with. As I see more examples and read up on the meaning of certain symbols I am getting better at making the transition. In the above program Susan guided me and informed me that % is used for modules.
-Samantha