Assignment: Elliptic Curve Cryptography, Due Monday, 11:59pm

Chris Betsill

November 7, 2016

i <-

1. Is this group cyclic?

Let E be the group defined by the elliptic curve $y^2 = x^3 - 3x + 3$ modulo 7.

a. List all the points in E. Explain how you know you have found all the points.

```
findPoints <- function(b, c, p){
  cat("NA, NA \n")
  for(x in 0:p-1){
    y2 <- mod.bigz(powm(x, 3, p) + mul.bigz(x, b) + c, p)
    if(y2 ==0)
      cat(x, ", 0 \n")
    if(1 == powm(y2,  div.bigz(sub.bigz(p, 1), 2), p)){
      y <- powm(y2, div.bigz(add.bigz(p, 1), 4), p)
      negy <- -y %% p
      cat(x, ",", as.integer(y), "\n")
      cat(x, ",", as.integer(negy), "\n")
    }

  }
}
findPoints(-3, 3, 7)
```

```
NA, NA
1 , 1
1 , 6
3 , 0
5 , 1
5 , 6
```

b. Show that the number of points in E satisifies Hasse's Theorem.

```
n <- 2*sqrt(7)+7+1
print(floor(n%%7))
```

```
[1] 6
```

From part a, you can see that there are 6 points - which according to Hasse's theorem (calculated above) is the maximum number of points on the Elliptical Curve.

c. Is the group E a cyclic group? Prove or disprove. If all orders were prime, then it would be cyclic

2. How about this one?

Let E be the group defined by the elliptic curve $y^2 = x^3 - x$ modulo 71.

a. Show that E contains exactly 72 points.

```
numPoints <- function(b, c, p){
  num <- 1
  for(x in 0:p-1){
    if(x > 0){
    y2 <- mod.bigz(powm(x, 3, p) + mul.bigz(x, b) + c, p)
    if(y2 == 0)
      num <- num + 1
    if(1 == powm(y2,  div.bigz(sub.bigz(p, 1), 2), p)){
      num <- num +2
    }

    }
  }
  return(num) # for some reason, the first point found has an x of -1... not sure why since x starts at
}
#findPoints(-1,0, 71)
numPoints(-1,0, 71)
```

```
[1] 71
```

b. Find all the different possible orders of elements in E.

```
#identity function runs for a while but then fails with Error in if (all(p1 == p2)) { : missing value w
# I spent a while chasing this and was unable to find the error. If this worked, you could find orders
identity <- function(b, c, p, point){
  i <- 1
  while(1){
    pt <- ecPowModp(b, c, p, point, i)
    print(pt)
    if(all(is.na(pt)))
      return(i)
    i <- i+1
  }
}
identity(-1, 0, 71, c(2,19))
findOrders <- function(b, c, p){
  xs <- vector()
  ys <- vector()
  orders <- vector()
  for(x in 0:p-1){
    if(x >= 0){
    y2 <- mod.bigz(powm(x, 3, p) + mul.bigz(x, b) + c, p)
    if(y2 == 0){
      xs <- append(xs, x)
      ys <- append(ys, 0)
    }
```

```r
    if(1 == powm(y2,  div.bigz(sub.bigz(p, 1), 2), p)){
      y <- powm(y2, div.bigz(add.bigz(p, 1), 4), p)
      negy <- -y %% p
      xs <- append(xs, x)
      ys <- append(ys, as.integer(y))
    }

    }
  }
  for(i in 1:length(xs))
    orders <- append(orders, identity(b, c, p, c(xs[i], ys[i])))
}
#findOrders(-1, 0, 71)
```

c. Is E cyclic? Prove or disprove.

If all orders were prime, then it would be cyclic

3. Find a point on a curve

Alice wants to send the message $m = 9230923203240394234$ using a cryptosystem based on the elliptic curve $y^2 = x^3 + 7x + 9$ modulo $p = 34588345934850984359911$.

a. Show that there is no point of the form (m, y) on this elliptic curve.

```r
m <- as.bigz("9230923203240394234")
p <- as.bigz("34588345934850984359911")
b <- 7
c <- 9

isPoint <- function(b, c, p, m){
  y2 <- mod.bigz(powm(m, 3, p)+mul.bigz(b, m)+c, p)
  if(1 == powm(y2,  div.bigz(sub.bigz(p, 1), 2), p)){
    y <- powm(y2, div.bigz(add.bigz(p, 1), 4), p)
    print("X is:")
    print(m)
    print("Y is:")
    print(y)
    return(TRUE)
  }

  return(FALSE)
}
isPoint(b,c,p,m)
```

```
[1] FALSE
```

b. Encode m as a point on this curve by adding a digit. That is, find a point of the form $(10m + k, y)$ on this curve, for some value of k between 0 and 9.

```r
for(i in 0:9){
  m2 <- add.bigz(i, mul.bigz(m, 10))
  if(isPoint(b,c,p,m2))
    break
}
```

```
[1] "X is:"
Big Integer ('bigz') :
[1] 92309232032403942343
[1] "Y is:"
Big Integer ('bigz') :
[1] 210561400666395769566 10
```

c. What is the approximate probability that this method (adding a single digit) will fail to produce a point on the curve?

Approximatly $.5^{10} = 0.0009765625$

4. Factor using an elliptic curve

Factor the number 2875605016366351 using Lenstra's method, with an elliptic curve of your choice. Use your `ecPowModp` function. Show your work.

E is $y^2 = x^3 + 5x - 5 \bmod n$

```r
#this code was used, and after the second (first was negative) printed statment I stopped it - I couldn

n <- as.bigz("2875605016366351")
run <- 1
fac <- function(b, c){
for(i in 1:15){
  tryCatch(ecPowModp(b, c, n, c(1,1), factorial(i)), warning=function(w){
    cat("x^3+",b,"x+", c,"at ", i)
    print(w)
  })
}
  fac(b+1, c-1)
}
fac(1, -1)

#x^3+48x-48 at  11
```

```r
gcd(429877887478171, 2875605016366351)
```

```
[1] 82351
```

5. Elliptic curve discrete logarithm

Let $G = (23, 14)$ be a point on the elliptic curve $y^2 = x^3 + 4x - 12063$ modulo 34543427. Find n such that $nG = (10735908, 411234)$.

```r
# This is me trying to implement a BSGS attack on a EC discreet log. It takes a long time to run..

# The one thing I wasnt sure on from the notes is if the two values are
g <- c(23, 14)
b <- as.bigz("4")
c <- as.bigz("-12063")
p <- as.bigz("34543427")
N <- 100000
ng <- c(as.bigz("10735908"), as.bigz("411234"))
vec1 <- vector()
vec2 <- vector()
for(j in 1:N){
  temp <- ecPowModp(b, c, p, g, j)
  append(vec1, temp)
  for(k in 1:N){
  temp2 <- ecPowModp(b, c, p, g, j)
  temp3 <- ecPowModp(b, c, p, temp2, N)
  temp4 <- ecAddModp(b, c, p, ng, ecNeg(temp3))
  vec2 <- vec2(vec2, temp4)
  if(length(intersect(vec1, vec2)) > 1){
    j <- which(vec1, intersect(vec1, vec2))
    k <-  which(vec2, intersect(vec1, vec2))
    return(j+N*k)
  }

  }
}
```

6. Elliptic curve ElGamal

Illustrate the ElGamal cryptosystem on the elliptic curve $y^2 = x^3 + 4x - 12063$ modulo 34543427. Let $\alpha = G = (23, 14)$.

a. Alice wants to send the message $m = 20161908$. Find a point $P_m = (m, y)$ on the curve, if possible. If no such point exists, pad m to obtain a point P_m on the curve.

```r
m <- as.bigz("20161908")
p <- as.bigz("34543427")
b <- 4
c <- -12063
pad <- 10

isPoint(b, c, p, m)
```

```
[1] "X is:"
Big Integer ('bigz') :
[1] 20161908
[1] "Y is:"
Big Integer ('bigz') :
[1] 31307119

[1] TRUE
```

b. For his private key, Bob chooses $a = 1945$. What information does Bob publish?

Bob publishes the elliptical curve E, a point α, and $\beta = 1945 * \alpha$

c. What message does Alice send?

Alice sends r which is eqqual to $r * \alpha$ in E where r is her secret random number, and t, which is equal to $m + k\beta$ in E

d. Show how Bob can decrypt the message.

Bob decrypts using his secret $a = 1945$ by computing $t - ar$ in E

7. Elliptic curve Diffie-Hellman

Illustrate the ellptic curve Diffie-Hellman key exchange on the elliptic curve $y^2 = x^3 + 4x - 12063$ modulo 34543427. Let $G = (23, 14)$. Suppose Alice's secret number is $N_A = 1984$, and Bob's secret number is $N_B = 2003$.

a. Compute the messages that Alice and Bob send to each other.

```r
na <- as.bigz("1984")
nb <- as.bigz("2003")
m <- as.bigz("20161908")
p <- as.bigz("34543427")
g <- c(23,14)
b <- 4
c <- -12063

ma <- ecPowModp(b, c, p, g, na)
ma
```

```
Big Integer ('bigz') object of length 2:
[1] 32476063 13213737
```

```r
mb <- ecPowModp(b, c, p, g, nb)
mb
```

```
Big Integer ('bigz') object of length 2:
[1] 19012677 5035018
```

b. Compute the shared key.

```r
ms <- ecPowModp(b, c, p, mb, na) # note this is the same as ecPowModp(b, c, p, ma, nb)
ms
```

```
Big Integer ('bigz') object of length 2:
[1] 24397553 13954137
```