Today.

Last time:
 Shared (and sort of kept) secrets.

Today: Errors
 Tolerate Loss: erasure codes.
 Tolerate corruption!

Poll

Line: $y = mx + b$

Poly: ”y” $= P(x) = a_d x^d + a_{d-1}x^{d-1}...a_0 x^0$

Everything below is true. Mark if you know it and perhaps why it is true.

(A) Two points determine a line: $mx + b$
(B) A root of $P(x)$, is a where $P(a) = 0$.
(C) A degree d polynomial has at most d roots.
(D) Arithmetic modulo a prime p is a ”field”.

(A) If a polynomial has a root at a, $P(x) = Q(x)(x-a)$.
(B) A line has at most one root, if not always zero.
(C) System: $y_1 = mx_1 + b, y_2 = mx_2 + b$ has unique solution $(m, b.)$
(D) Degree of a polyomial $P(x)^2$ is $2d$ if $P(x)$ is degree d.

(C) may not be true.

The mathematics.

There is a unique polynomial of degree d that contains any $d+1$ points.

Assumption: a field, in particular, arithmetic $\mod p$.

Big Idea:

 A polynomial: $P(x) = a_d x^d + \cdots a_0$ has $d+1$ coefficients.
 Any set of $d+1$ points determines the polynomial.

Stare at the above. What does it mean?
 Many representations of a polynomial!
  One coefficient represention.
  Many, many point,value representations.

Some details:
  Degree d generally degree “at most” d.
   (example: choose 10 points on a line.)
  Arithmetic $\pmod p \implies$ work with $O(\log p)$ bit numbers.

Proof sketches.

Property 2 A polynomial: $P(x) = a_d x^d + \cdots a_0$ has $d+1$ coefficients.
 Any set of $d+1$ points uniquely determines the polynomial.

Existence: Lagrange Intropolation.
 Degree d, $\Delta_i(x)$ polynomials.
   factors of $(x - x_j)$ to zero out at $x_j \neq x_i$.
    Multiply by zero. My love is won.
 Combine.

Uniqueness:
 Property 1 A non-zero degree d polynomial has at most d roots.
   Factoring: $P(x)$ with roots $r_1, \ldots, r_d$
    $\implies P(x) = c(x - r_0)(x - r_1)\ldots(x - r_d)$.

 Love me some contradiction!
  Two polynomials: $P(x), Q(x)$, $P(x) - Q(x)$ has too many roots.

Finite Fields

Proof works for reals, rationals, and complex numbers.

..but not for integers, since no multiplicative inverses.

Arithmetic modulo a prime p has multiplicative inverses..

..and has only a finite number of elements.

Good for computer science.

Arithmetic modulo a prime m is a **finite field** denoted by F_m or $GF(m)$.

Intuitively, a field is a set with operations corresponding to addition, multiplication, and division.

Secret Sharing

Modular Arithmetic Fact: Exactly one polynomial degree $\leq d$ over $GF(p)$, $P(x)$, that hits $d+1$ points.

Shamir's k out of n Scheme:
Secret $s \in \{0, \ldots, p-1\}$

1. Choose $a_0 = s$, and randomly $a_1, \ldots, a_{k-1}$.
2. Let $P(x) = a_{k-1}x^{k-1} + a_{k-2}x^{k-2} + \cdots a_0$ with $a_0 = s$.
3. Share i is point $(i, P(i) \mod p)$.

Roubustness: Any k knows secret.
Knowing k pts, only one $P(x)$, evaluate $P(0)$.
Secrecy: Any $k-1$ knows nothing.
Knowing $\leq k-1$ pts, any $P(0)$ is possible.

Two points make a line: the value of one point allows any y-intercept.

 3 kids hand out 3 points. Any two know the line.

Minimality.

Need $p > n$ to hand out n shares: $P(1) \ldots P(n)$.

For b-bit secret, must choose a prime $p > 2^b$.

Theorem: There is always a prime between n and $2n$.
Chebyshev said it,
And I say it again,
There is always a prime
Between n and 2n.

Working over numbers within 1 bit of secret size. **Minimality.**

With k shares, reconstruct polynomial, $P(x)$.

With $k-1$ shares, any of p values possible for $P(0)$!

(Almost) any b-bit string possible!

(Almost) the same as what is missing: one $P(i)$.

Runtime.

Runtime: polynomial in k, n, and $\log p$.

1. Evaluate degree $k-1$ polynomial n times using $\log p$-bit numbers.
2. Reconstruct secret by solving system of k equations using $\log p$-bit arithmetic.

A bit more counting.

What is the number of degree d polynomials over $GF(m)$?

- m^{d+1}: $d+1$ coefficients from $\{0,\ldots,m-1\}$.
- m^{d+1}: $d+1$ points with y-values from $\{0,\ldots,m-1\}$

Infinite number for reals, rationals, complex numbers!

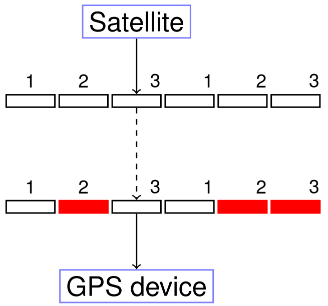
Secret Sharing.

n people, k is enough.

(A) The modulus needs to be at least $n+1$.
(B) The modulus needs to be at least k.
(C) Use degree k polynomial, hand out n points.
(D) Use degree n polynomial, hand out k points.
(E) Use degree $k-1$ polynomial, hand out n points.
(F) The modulus needs to be at least 2^s, where s is value of secret.
(G) The modulus needs to be at least 2^s, where s is size of secret.

(A), (B), (E), (F)

Erasure Codes.



3 packet message. So send 6!

Lose 3 out 6 packets.

Gets packets 1,1,and 3.

Solution Idea.

n packet message, channel that loses k packets.

Must send $n+k$ packets!

Any n packets should allow reconstruction of n packet message.

Any n point values allow reconstruction of degree $n-1$ polynomial.

Alright!!!!!!

Use polynomials.

The Scheme

Problem: Want to send a message with n packets.

Channel: Lossy channel: loses k packets.

Question: Can you send $n+k$ packets and recover message?

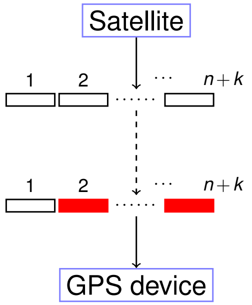
A degree $n-1$ polynomial determined by any n points!

Erasure Coding Scheme: message = $m_0, m_1 \ldots, m_{n-1}$.

1. Choose prime $p \approx 2^b$ for packet size b.
2. $P(x) = m_{n-1}x^{n-1} + \cdots m_0 \pmod{p}$.
3. Send $P(1), \ldots, P(n+k)$.

Any n of the $n+k$ packets gives polynomial ...and message!

Erasure Codes.



n packet message. So send $n+k$!

Lose k packets.

Any n packets is enough!

n packet message.

Optimal.

Information Theory.

Size: Can choose a prime between 2^{b-1} and 2^b.
(Lose at most 1 bit per packet.)

But: packets need label for x value.

There are Galois Fields $GF(2^n)$ where one loses nothing.

– Can also run the Fast Fourier Transform.

In practice, $O(n)$ operations with almost the same redundancy.

Comparison with Secret Sharing: information content.

Secret Sharing: each share is size of whole secret.
Coding: Each packet has size $1/n$ of the whole message.

Erasure Code: Example.

Send message of 1,4, and 4.

Make polynomial with $P(1) = 1$, $P(2) = 4$, $P(3) = 4$.

How?

Lagrange Interpolation.
Linear System.

Work modulo 5.

$P(x) = x^2 \pmod{5}$
$P(1) = 1, P(2) = 4, P(3) = 9 = 4 \pmod{5}$

Send $(0, P(0)) \ldots (5, P(5))$.

6 points. Better work modulo 7 at least!

Why? $(0, P(0)) = (5, P(5)) \pmod{5}$

Example

Make polynomial with $P(1) = 1$, $P(2) = 4$, $P(3) = 4$.

Modulo 7 to accommodate at least 6 packets.

Linear equations:

$$\begin{aligned} P(1) &= a_2 + a_1 + a_0 &\equiv 1 \pmod{7} \\ P(2) &= 4a_2 + 2a_1 + a_0 &\equiv 4 \pmod{7} \\ P(3) &= 2a_2 + 3a_1 + a_0 &\equiv 4 \pmod{7} \end{aligned}$$

$6a_1 + 3a_0 = 2 \pmod{7}$, $5a_1 + 4a_0 = 0 \pmod{7}$

$a_1 = 2a_0$. $a_0 = 2 \pmod{7}$ $a_1 = 4 \pmod{7}$ $a_2 = 2 \pmod{7}$

$P(x) = 2x^2 + 4x + 2$

$P(1) = 1$, $P(2) = 4$, and $P(3) = 4$

Send
Packets: $(1,1), (2,4), (3,4), (4,7), (5,2), (6,0)$

Notice that packets contain “x-values”.

Bad reception!

Send: $(1,1), (2,4), (3,4), (4,7), (5,2), (6,0)$

Recieve: (1,1) (2,4), (6,0)
Reconstruct?

Format: $(i, R(i))$.

Lagrange or linear equations.

$$\begin{aligned} P(1) &= a_2 + a_1 + a_0 &\equiv 1 \pmod{7} \\ P(2) &= 4a_2 + 2a_1 + a_0 &\equiv 4 \pmod{7} \\ P(6) &= 2a_2 + 3a_1 + a_0 &\equiv 0 \pmod{7} \end{aligned}$$

Channeling Sahai ...

$P(x) = 2x^2 + 4x + 2$

Message? $P(1) = 1, P(2) = 4, P(3) = 4$.

Questions for Review

You want to encode a secret consisting of 1,4,4.

How big should modulus be?
Larger than 144 and prime!

Remember the secret, $s = 144$, must be one of the possible values.

You want to send a message consisting of packets 1,4,2,3,0

through a noisy channel that loses 3 packets.

How big should modulus be?
Larger than 8 and prime!

The other constraint: arithmetic system can represent $0,1,2,3,4$.

Send n packets b-bit packets, with k errors.
Modulus should be larger than $n+k$ and also larger than 2^b.

Polynomials.

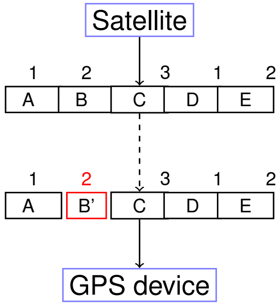
- ..give Secret Sharing.
- ..give Erasure Codes.

Error Correction:

Noisy Channel: corrupts k packets. (rather than loss.)

Additional Challenge: Finding which packets are corrupt.

Error Correction



3 packet message. Send 5.

Corrupts 1 packets.

The Scheme.

Problem: Communicate n packets $m_1,\ldots,m_n$
on noisy channel that corrupts $\le k$ packets.

Reed-Solomon Code:

1. Make a polynomial, $P(x)$ of degree $n-1$,
   that encodes message.
   - $P(1) = m_1,\ldots,P(n) = m_n$.
   - Comment: could encode with packets as coefficients.
2. Send $P(1),\ldots,P(n+2k)$.

After noisy channel: Recieve values $R(1),\ldots,R(n+2k)$.

Properties:
(1) $P(i) = R(i)$ for at least $n+k$ points i,
(2) $P(x)$ is unique degree $n-1$ polynomial
that contains $\ge n+k$ received points.

Properties: proof.

$P(x)$: degree $n-1$ polynomial.
Send $P(1),\ldots,P(n+2k)$
Receive $R(1),\ldots,R(n+2k)$
At most k i's where $P(i) \ne R(i)$.

Properties:
(1) $P(i) = R(i)$ for at least $n+k$ points i,
(2) $P(x)$ is unique degree $n-1$ polynomial
that contains $\ge n+k$ received points.

Proof:
(1) Sure. Only k corruptions.
(2) Degree $n-1$ polynomial $Q(x)$ consistent with $n+k$ points.
$Q(x)$ agrees with $R(i)$, $n+k$ times.
$P(x)$ agrees with $R(i)$, $n+k$ times.

Total points contained by both	: $2n+2k$.	P	Pigeons.
Total points to choose from	: $n+2k$.	H	Holes.
Points contained by both	: $\ge n$.	$\ge P-H$	Collisions.

$\implies Q(i) = P(i)$ at n points.
$\implies Q(x) = P(x)$. □

Example.

Message: $3,0,6$.

Reed Solomon Code: $P(x) = x^2+x+1 \pmod 7$ has
$P(1) = 3, P(2) = 0, P(3) = 6$ modulo 7.

Send: $P(1) = 3, P(2) = 0, P(3) = 6, P(4) = 0, P(5) = 3$.

(Aside: Message in plain text!)

Receive $R(1) = 3, R(2) = 1, R(3) = 6, R(4) = 0, R(5) = 3$.

$P(i) = R(i)$ for $n+k = 3+1 = 4$ points.

Slow solution.

Brute Force:
For each subset of $n+k$ points
  Fit degree $n-1$ polynomial, $Q(x)$, to n of them.
  Check if consistent with $n+k$ of the total points.
  If yes, output $Q(x)$.

- For subset of $n+k$ pts where $R(i) = P(i)$, method will reconstruct $P(x)$!
- For any subset of $n+k$ pts,
  1. there is unique degree $n-1$ polynomial $Q(x)$ that fits n of them
  2. and where $Q(x)$ is consistent with $n+k$ points
     $\Longrightarrow P(x) = Q(x)$.

Reconstructs $P(x)$ and only $P(x)$!!

Example.

Received $R(1) = 3, R(2) = 1, R(3) = 6, R(4) = 0, R(5) = 3$

Find $P(x) = p_2x^2 + p_1x + p_0$ that contains $n+k = 3+1$ points.

All equations..

$$
\begin{aligned}
p_2 + p_1 + p_0 &\equiv 3 \pmod 7\\
4p_2 + 2p_1 + p_0 &\equiv 1 \pmod 7\\
2p_2 + 3p_1 + p_0 &\equiv 6 \pmod 7\\
2p_2 + 4p_1 + p_0 &\equiv 0 \pmod 7\\
4p_2 + 5p_1 + p_0 &\equiv 3 \pmod 7
\end{aligned}
$$

Assume point 1 is wrong and solve..no consistent solution!
Assume point 2 is wrong and solve...consistent solution!

In general..

$P(x) = p_{n-1}x^{n-1} + \cdots p_0$ and receive $R(1), \ldots R(m = n+2k)$.

$$
\begin{aligned}
p_{n-1} + \cdots p_0 &\equiv R(1) \pmod p\\
p_{n-1}2^{n-1} + \cdots p_0 &\equiv R(2) \pmod p\\
&\cdot\\
p_{n-1}i^{n-1} + \cdots p_0 &\equiv R(i) \pmod p\\
&\cdot\\
p_{n-1}(m)^{n-1} + \cdots p_0 &\equiv R(m) \pmod p
\end{aligned}
$$

Error!! .... Where???
Could be anywhere!!! ...so try everywhere.
Runtime: $\binom{n+2k}{k}$ possibilitities.

Something like $(n/k)^k$ ...Exponential in k!.

How do we find where the bad packets are efficiently?!?!?!

Ditty...

Oh where, Oh where
has my little dog gone?
Oh where, oh where can he be

With his ears cut short
And his tail cut long
Oh where, oh where can he be?

Oh where, Oh where
have my packets gone.. wrong?
Oh where, oh where do they not fit.

With the polynomial well put
But the channel a bit wrong
Where, oh where do we look?

Where oh where can my bad packets be?

$$
\begin{aligned}
E(1)(p_{n-1} + \cdots p_0) &\equiv R(1)E(1) \pmod p\\
\mathbf{0} \times E(2)(p_{n-1}2^{n-1} + \cdots p_0) &\equiv R(2)E(2) \pmod p\\
&\vdots\\
E(m)(p_{n-1}(m)^{n-1} + \cdots p_0) &\equiv R(n+2k)E(m) \pmod p
\end{aligned}
$$

Idea: Multiply equation i by 0 if and only if $P(i) \neq R(i)$.
  Zero times anything is zero!!!!! My love is won.
  All equations satisfied!!!!!

But which equations should we multiply by 0? Where oh where...??

We will use a polynomial!!! That we don't know. But can find!

Errors at points $e_1, \ldots, e_k$. (In diagram above, $e_1 = 2$.)

Error locator polynomial: $E(x) = (x - e_1)(x - e_2)\ldots(x - e_k)$.

$E(i) = 0$ if and only if $e_j = i$ for some j

Multiply equations by $E(\cdot)$. (Above E(x) = (x-2).)

All equations satisfied!!

Example.

Received $R(1) = 3, R(2) = 1, R(3) = 6, R(4) = 0, R(5) = 3$

Find $P(x) = p_2x^2 + p_1x + p_0$ that contains $n+k = 3+1$ points.

Plugin points...

$$
\begin{aligned}
(1-2)(p_2 + p_1 + p_0) &\equiv (3)(1-2) \pmod 7\\
(2-2)(4p_2 + 2p_1 + p_0) &\equiv (1)(2-2) \pmod 7\\
(3-2)(2p_2 + 3p_1 + p_0) &\equiv (6)(3-2) \pmod 7\\
(4-2)(2p_2 + 4p_1 + p_0) &\equiv (0)(4-2) \pmod 7\\
(5-2)(4p_2 + 5p_1 + p_0) &\equiv (3)(5-2) \pmod 7
\end{aligned}
$$

Error locator polynomial: $(x-2)$.

Multiply equation i by $(i-2)$. All equations satisfied!

But don't know error locator polynomial! Do know form: $(x - e)$.

4 unknowns (p_0, p_1, p_2 and e), 5 nonlinear equations.

..turn their heads each day,

$$\begin{aligned} E(1)(p_{n-1}+\cdots p_0) &\equiv R(1)E(1) \pmod{p} \\ &\vdots \\ E(i)(p_{n-1}i^{n-1}+\cdots p_0) &\equiv R(i)E(i) \pmod{p} \\ &\vdots \\ E(m)(p_{n-1}(n+2k)^{n-1}+\cdots p_0) &\equiv R(m)E(m) \pmod{p} \end{aligned}$$

...so satisfied, I'm on my way.

$m = n+2k$ satisfied equations, $n+k$ unknowns. But nonlinear!

Let $Q(x) = E(x)P(x) = a_{n+k-1}x^{n+k-1}+\cdots a_0$.

Equations:

$$Q(i) = R(i)E(i).$$

and linear in a_i and coefficients of $E(x)$!

Finding $Q(x)$ and $E(x)$?

- $E(x)$ has degree k ...

  $$E(x) = x^k + b_{k-1}x^{k-1}\cdots b_0.$$

  $\Longrightarrow$ k (unknown) coefficients. Leading coefficient is 1.
- $Q(x) = P(x)E(x)$ has degree $n+k-1$ ...

  $$Q(x) = a_{n+k-1}x^{n+k-1} + a_{n+k-2}x^{n+k-2} + \cdots a_0$$

  $\Longrightarrow$ $n+k$ (unknown) coefficients.

Number of unknown coefficients: $n+2k$.

Solving for $Q(x)$ and $E(x)$...and $P(x)$

For all points $1,\ldots,i,n+2k = m$,

$$Q(i) = R(i)E(i) \pmod{p}$$

Gives $n+2k$ linear equations.

$$\begin{aligned} a_{n+k-1}+\ldots a_0 &\equiv R(1)(1+b_{k-1}\cdots b_0) \pmod{p} \\ a_{n+k-1}(2)^{n+k-1}+\ldots a_0 &\equiv R(2)((2)^k+b_{k-1}(2)^{k-1}\cdots b_0) \pmod{p} \\ &\vdots \\ a_{n+k-1}(m)^{n+k-1}+\ldots a_0 &\equiv R(m)((m)^k+b_{k-1}(m)^{k-1}\cdots b_0) \pmod{p} \end{aligned}$$

..and $n+2k$ unknown coefficients of $Q(x)$ and $E(x)$!

Solve for coefficients of $Q(x)$ and $E(x)$.

Find $P(x) = Q(x)/E(x)$.

Example.

Received $R(1) = 3, R(2) = 1, R(3) = 6, R(4) = 0, R(5) = 3$

$Q(x) = E(x)P(x) = a_3x^3 + a_2x^2 + a_1x + a_0$

$E(x) = x - b_0$

$Q(i) = R(i)E(i)$.

$$\begin{aligned} a_3+a_2+a_1+a_0 &\equiv 3(1-b_0) \pmod{7} \\ a_3+4a_2+2a_1+a_0 &\equiv 1(2-b_0) \pmod{7} \\ 6a_3+2a_2+3a_1+a_0 &\equiv 6(3-b_0) \pmod{7} \\ a_3+2a_2+4a_1+a_0 &\equiv 0(4-b_0) \pmod{7} \\ 6a_3+4a_2+5a_1+a_0 &\equiv 3(5-b_0) \pmod{7} \end{aligned}$$

$a_3 = 1$, $a_2 = 6$, $a_1 = 6$, $a_0 = 5$ and $b_0 = 2$.

$Q(x) = x^3+6x^2+6x+5$.

$E(x) = x-2$.

Example: finishing up.

$Q(x) = x^3+6x^2+6x+5$.
$E(x) = x-2$.

```
                1 x^2 + 1 x + 1
           ----------------
  x - 2 ) x^3  + 6 x^2 + 6 x + 5
          x^3  - 2 x^2
          ----------
                1 x^2 + 6 x + 5
                1 x^2 - 2 x
                ---------------
                          x + 5
                          x - 2
                          -----
                              0
```

$P(x) = x^2+x+1$
Message is $P(1) = 3, P(2) = 0, P(3) = 6$.

What is $\frac{x-2}{x-2}$? 1
Except at $x = 2$? Hole there?

Error Correction: Berlekamp-Welsh

Message: $m_1,\ldots,m_n$.
Sender:

1. Form degree $n-1$ polynomial $P(x)$ where $P(i) = m_i$.
2. Send $P(1),\ldots,P(n+2k)$.

Receiver:

1. Receive $R(1),\ldots,R(n+2k)$.
2. Solve $n+2k$ equations, $Q(i) = E(i)R(i)$ to find $Q(x) = E(x)P(x)$ and $E(x)$.
3. Compute $P(x) = Q(x)/E(x)$.
4. Compute $P(1),\ldots,P(n)$.

Check your undersanding.

You have error locator polynomial!

Where oh where have my packets gone wrong?

Factor? Sure.
Check all values? Sure.

Efficiency? Sure. Only $n+2k$ values.
See where it is 0.

Hmmm...

Is there one and only one $P(x)$ from Berlekamp-Welsh procedure?

Existence: there is a $P(x)$ and $E(x)$ that satisfy equations.

Unique solution for $P(x)$

Uniqueness: any solution $Q'(x)$ and $E'(x)$ have

$$\frac{Q'(x)}{E'(x)} = \frac{Q(x)}{E(x)} = P(x). \qquad (1)$$

Proof:
We claim

$$Q'(x)E(x) = Q(x)E'(x) \text{ on } n+2k \text{ values of } x. \qquad (2)$$

Equation 2 implies 1:

$Q'(x)E(x)$ and $Q(x)E'(x)$ are degree $n+2k-1$
and agree on $n+2k$ points
$E(x)$ and $E'(x)$ have at most k zeros each.
Can cross divide at n points.
$\implies \frac{Q'(x)}{E'(x)} = \frac{Q(x)}{E(x)}$ equal on n points.
Both degree $\leq n \implies$ Same polynomial! □

Last bit.

Fact: $Q'(x)E(x) = Q(x)E'(x)$ on $n+2k$ values of x.

Proof: Construction implies that

$$Q(i) = R(i)E(i)$$
$$Q'(i) = R(i)E'(i)$$

for $i \in \{1, \ldots n+2k\}$.

If $E(i) = 0$, then $Q(i) = 0$. If $E'(i) = 0$, then $Q'(i) = 0$.
$\implies Q(i)E'(i) = Q'(i)E(i)$ holds when $E(i)$ or $E'(i)$ are zero.

When $E'(i)$ and $E(i)$ are not zero

$$\frac{Q'(i)}{E'(i)} = \frac{Q(i)}{E(i)} = R(i).$$

Cross multiplying gives equality in fact for these points. □

Points to polynomials, have to deal with zeros!

Example: dealing with $\frac{x-2}{x-2}$ at $x = 2$.

Yaay!!

Berlekamp-Welsh algorithm decodes correctly when k errors!

Poll

Say you sent a message of length 4, encoded as $P(x)$ where one sends packets $P(1), \ldots P(8)$.

You recieve packets $R(1), \ldots R(8)$.

Packets 1 and 4 are corrupted.

(A) $R(1) \neq P(1)$
(B) The degree of $P(x)E(x) = 3+2 = 5$.
(C) The degree of $E(x)$ is 2.
(D) The number of coefficients of $P(x)$ is 4.
(E) The number of coefficients of $P(x)Q(x)$ is 6.

all true.

(A) $E(x) = (x-1)(x-4)$
(B) The number of coefficents in $E(x)$ is 2.
(C) The number of unknown coefficents in $E(x)$ is 2.
(D) $E(x) = (x-1)(x-2)$
(E) $R(4) \neq P(4)$
(F) The degree of $R(x)$ is 5.

(A), (C), (E).

Summary. Error Correction.

Communicate n packets, with k erasures.

- How many packets? $n+k$
- How to encode? With polynomial, $P(x)$.
- Of degree? $n-1$
- Recover? Reconstruct $P(x)$ with any n points!

Communicate n packets, with k errors.

- How many packets? $n+2k$
  - Why?
    - k changes to make diff. messages overlap
- How to encode? With polynomial, $P(x)$. Of degree? $n-1$.
- Recover?
  - Reconstruct error polynomial, $E(X)$, and $P(x)$!
    - Nonlinear equations.
- Reconstruct $E(x)$ and $Q(x)=E(x)P(x)$. Linear Equations.
  - Polynomial division! $P(x)=Q(x)/E(x)$!

Reed-Solomon codes. Welsh-Berlekamp Decoding. Perfection!

Cool.

Really Cool!