ElGamal Bob chooses

- large p, primitive root $g \pmod p$
- random x, $h := g^x \pmod p$

public key:	private key:
p, g, h	x

A encrypts: $c = (c_1, c_2)$

$c_1 = g^y$

$c_2 = h^y m \pmod p$ y selected randomly

B decrypts: $m = c_2 c_1^{-x} \pmod p$

COMMENTS:

- Given p, how many choices for g?
  # primitive roots mod p $= \phi(\phi(p)) = \phi(p-1)$
  EG $p = 23$ $\phi(22) = \phi(2)\phi(11) = 10$ choices for g
- Does A need to select new y each time?
  yes! $(c_1, c_2) = (g^y, h^y m)$, $(\tilde{c}_1, \tilde{c}_2) = (g^y, h^y \tilde{m})$ $\Rightarrow c_2/\tilde{c}_2 = m/\tilde{m}$
- obtaining m from c as hard as:
  CDH: given $g, g^x, g^y \pmod p$, find $g^{xy} \pmod p$
  decisional DDH: given $g, g^x, g^y, r \pmod p$, decide whether $g^{xy} = r \pmod p$ still hard?
- p often chosen to be "safe prime"
  DEF: p, $\frac{p-1}{2}$ both prime
  $\phi(p) = p-1 = 2 \cdot \frac{p-1}{2}$
  $\Rightarrow$ possible orders: 1, 2, $\frac{p-1}{2}$, $p-1$ (primitive roots)
  1, -1 ($x^2 \equiv 1$)