

```

> a1 := x^4+8*x^2+6*x+8;
a2 := x^6+3*x^5-x^4+2*x^3-3*x+3;
a3 := x^8+x^7+x^6+2*x^4+5*x^3+2*x^2+8;
      a1 :=  $x^4 + 8x^2 + 6x + 8$ 
      a2 :=  $x^6 + 3x^5 - x^4 + 2x^3 - 3x + 3$ 
      a3 :=  $x^8 + x^7 + x^6 + 2x^4 + 5x^3 + 2x^2 + 8$ 

```

(1)

```

> `mod` := mods;
      mod := mods

```

(2)

```

> my_split := proc(g, k, p)
  local h, gh, rnd_poly, w, a, b;
  if g = 0 then return 1; fi;
  if degree(g) = k then return g; fi;
  while true do
    rnd_poly := randpoly(x, degree=k) mod p;
    w := Powmod(rnd_poly, (p^k-1)/2, g, x);
    h := Gcd(g, w-1) mod p;
    if h <> 1 and h <> g then break; fi;
  od;
  gh := Quo(g, h, x) mod p;
  a := my_split(h, k, p);
  b := my_split(gh, k, p);
  return a*b;
end:
> my_factor := proc(a, p)
  local s, k, w, gk, result;
  s := a;
  result := 1;
  k := 1;
  w := x;
  while k <= floor(degree(s)/2) do
    w := Rem(Powmod(w, p, s, x) mod p, s, x) mod p;
    gk := Gcd(w - x, s) mod p;
    if degree(gk) >= k then
      result := result * my_split(gk, k, p);
    fi;
    s := Quo(s, gk, x) mod p;
    k := k + 1;
  od;
  if s <> 1 then result := result*s; fi;
  return result;
end:

```

```

> my_factor(a1, 11);
Factor(a1) mod 11;
      (x+2) (x+5) (x+1) (x+3)
      (x+2) (x+5) (x+1) (x+3)

```

(3)

```

> my_factor(a2, 11);
Factor(a2) mod 11;
      (x+2) (x^5+x^4-3x^3-3x^2-5x-4)
      (x+2) (x^5+x^4-3x^3-3x^2-5x-4)

```

(4)

```

> my_factor(a3, 11);
Factor(a3) mod 11;

```

$$(x^2 + x + 1) (x^3 + 3x + 5) (x^3 - 3x - 5)$$

$$(x^2 + x + 1) (x^3 + 3x + 5) (x^3 - 3x - 5)$$

(5)

> large_p := 10^20+ 129;

$$large_p := 1000000000000000000129$$

(6)

> a4 := x^2 - 3;
my_factor(a4, large_p);

$$a4 := x^2 - 3$$

$$(x - 28287745671504160848) (x + 28287745671504160848)$$

(7)

> a5 := x^2 - 5;
my_factor(a5, large_p);

$$a5 := x^2 - 5$$

$$(x + 14339274750131571137) (x - 14339274750131571137)$$

(8)

> a6 := x^2 - 7;
my_factor(a6, large_p);

$$a6 := x^2 - 7$$

$$x^2 - 7$$

(9)