EÖTVÖS LORÁND TUDOMÁNY EGYETEM
TERMÉSZETTUDOMÁNYI KAR

Versenyfeladatok az általános- és középiskolában

Szakdolgozat

Domán Dániel Gergő
Matematika BSC
Alkalmazott matematikus szakirány

Témavezető:
Dr. Kiss Emil
Algebra és Számelmélet tanszék
Eötvös Loránd Tudományegyetem
Természettudományi Kar

Köszönetnyilvánítás

Szeretnék köszönetet mondani Kiss Emil Tanár Úrnak, hogy elvállalta a témavezetést, valamint szaktudásával, tanácsaival és ötleteivel segítette a szakdolgozat elkészítését.

1. Előszó

A szakdolgozat célja a 2019-es OKTV döntő Matematika III. kategóriájának egyik feladatát és megoldásait bemutatni, ezzel kapcsolatban pedig egyéb kérdéseket feltenni, megválaszolni, amit pedig nem sikerült matematikai eszközökkel belátni, arra programot írni. A Kömalban várhatóan feladatként meg fog jelenni az egyik tétel valamely verziója. A szakdolgozatban szereplő bizonyítások részben a témavezetővel közös eredmények, részben Károlyi Gyula eredményei.

2. A feladat

Legyen f egész együtthatós polinom, $k \geq 2$ egész, és p prímszám. Tegyük fel, hogy az $f(0), f(1), ..., f(p-1)$ számok p-vel osztva k különböző maradékot adnak. Bizonyítsuk be, hogy ekkor f foka legalább $(p-1)/(k-1)$.

3. Megoldás I.

Legyen először $k = 2$, vagyis az $f(0), f(1), ..., f(p-1)$ számok p szerint pontosan 2 maradékosztályhoz tartoznak. Be kell látni, hogy f foka legalább $p-1$.

Az általánosság megszorítása nélkül feltehető, hogy az egyik maradékosztály a 0, mivel a konstans tagot tetszőlegesen módosíthatjuk. A másik maradékosztályt jelöljük d-vel.

A $0 \leq x \leq p-1$ egészek közül legyenek a_1, a_2,..., a_m azok, amelyekre $f(x)$ maradéka d-vel egyenlő modulo p. Ekkor tehát $0 < m < p$, és az a_i-ktől különböző 0 és $p-1$ közötti x egész számokra $f(x)$ p-vel osztva 0-át ad maradékul.

1. Tétel (kis-Fermat): p legyen prím, a pedig egész és nem egész számú többszöröse p-nek, ekkor $a^{p-1} \equiv 1 \; mod \; p$

A kis-Fermat-tétel alapján a $0 \leq x \leq p-1$ egészekre az $(x-a_i)^{p-1}$ hatvány 1 maradékot ad p-vel osztva, ha $x \neq a_i$, és persze 0, ha $x = a_i$. Ezért az

$$(1-(x-a_1)^{p-1}) + (1-(x-a_2)^{p-1}) + ... + (1-(x-a_m)^{p-1})$$

összegben ha $x = a_i$, egyedül az i-edik tag ad 1 maradékot modulo p, és az összes többi tag p-vel osztható, ha pedig x az a_i-k mindegyikétől különböző egész szám 0 és $p-1$ között, akkor mindegyik tag p-vel osztható. Tekintsük a

$$g(x) = d\sum_{i=1}^{n}(1-(x-a_i)^{p-1}$$

egész együtthatós $(p-1)$-ed fokú polinomot. Az előzőek alapján minden $0 \leq x \leq p-1$ egész szám esetében $g(x)$-nek ugyanannyi a p szerinti maradéka, mint $f(x)$-nek. Nyilván ekkor ugyanez igaz minden x egész számra is.

A $h = f - g$ polinomnak így minden egész helyen vett helyettesítési értéke p-vel osztható. A következő, egész együtthatós polinomokra vonatkozó segédtételt alkalmazzuk:

2. Tétel[1]**:** Ha adott két legfeljebb n-edfokú polinom $\mathbf{Z}_p$ fölött, amelyek több mint n helyen megegyeznek, akkor a két polinom egyenlő. ($\mathbf{Z}_p$ definícióját lásd később, a 6. fejezetben)

Lemma: Ha egy polinom foka n, ahol $n < p$, és legalább $n+1$ különböző p szerinti maradékosztályhoz tartozó egész szám behelyettesítésekor kapunk p-vel osztható értéket, akkor az összes együttható osztható p-vel.

Bizonyítás: A lemmát átfogalmazva $\mathbf{Z}_p$ fölötti polinomokra, ez azt jelenti, hogy legalább $n+1$ helyen 0-t vesz fel, azaz megegyezik az azonosan 0 polinommal. A 2. tétel miatt csak az azonosan 0 polinom lehet.

Ebből következik, hogy ha egy a p prímszámnál alacsonyabb fokú egész együtthatós polinomnak minden egész számnál vett helyettesítési értéke osztható p-vel, akkor a polinom mindegyik együtthatója osztható p-vel.

A g polinom $p-1$-edfokú, és a főegyütthatója $-md$, ami nem osztható p-vel. Ha f ennél alacsonyabb fokú volna, akkor h is $p-1$-edfokú volna ugyanazzal a főegyütthatóval, mint g, ami a segédtétel miatt lehetetlen. Tehát f foka valóban legalább $p-1$.

Rátérünk a feladat általános esetére. Legyenek az f polinom $0, 1, ..., p-1$ helyeken felvett értékeinek modulo p páronként különböző maradékai a $c_1, c_2, ..., c_k$ számok, és tekintsük az

$$F(x) = (f(x) - c_2)...(f(x) - c_k)$$

egész együtthatós polinomot. Nyilván F foka az f fokának $(k-1)$-szerese, tehát azt kell belátnunk, hogy F foka legalább $p-1$.

Legyen $0 \leq x \leq p-1$ tetszőleges egész. Ha $f(x)$ értéke c_1 modulo p, akkor $F(x)$-nek és $(c_1 - c_2)...(c_1 - c_k)$-nak ugyanannyi a p szerinti maradéka, és az utóbbi nyilván nem osztható p-vel. Ha pedig $f(x)$ maradéka p szerint valamelyik másik c_j -vel egyenlő, akkor viszont $p|F(x)$. Tehát az F polinomra teljesülnek a feladat feltételei $k = 2$-vel, így a korábban bizonyítottak szerint F foka valóban legalább $p-1$.

4. Megoldás 2.[2]

A fenti jelölésekkel $k \geq 2$ esetén

$$n \geq \frac{p-1}{k-1}.$$

3. Tétel: Ha $n < p-1$, akkor $f(0) + f(1) + \ldots + f(p-1) = 0$

Bizonyítás: Vegyük észre, hogy

$$\ell_b(x) = 1 - (x-b)^{p-1}$$

a b-hez tartozó Lagrange interpolációs alappolinom, ezért

$$f(x) = \sum_{b=0}^{p-1} f(b)\ell_b(x)\,.$$

Ebben $n < p-1$ miatt x^{p-1} együtthatója nulla, és ez $-f(0) - \ldots - f(p-1)$. A 3. tételt beláttuk.

Legyen f értékkészlete $a_1, \ldots, a_k \in \mathbf{Z_p}$, ahol az a_i multiplicitása $x_i \geq 1$. Nyilván $x_1 + \ldots + x_k = p$. Ha $nj < p-1$ valamilyen $1 \leq j$ egészre, akkor a 3. tételt az f^j polinomra alkalmazva

$$x_1 a_1^j + \ldots + x_k a_k^j = 0\,.$$

Tegyük fel indirekt, hogy $n(k-1) < p-1$. Ekkor a fenti egyenlőséget $0 \leq j \leq k-1$-re felírva homogén lineáris egyenletrendszert kapunk az x_i ismeretlenekre $\mathbf{Z_p}$ fölött, melynek determinánsa Vandermonde-féle, és így nem nulla. Ezért az egyenletrendszernek csak triviális megoldása van, azaz mindegyik x_i multiplicitás p-vel osztható. Ez $k > 1$ miatt lehetetlen.

5. Éles-e a becslés?

Fő kérdésünk, amit vizsgálunk, hogy éles-e ez a becslés, mikor igen, mikor nem.

Állítás: Ha $k = 2$, akkor éles. Vagyis van olyan $p-1$-ed fokú polinom, aminek az egész helyeken vett helyettesítési értékei p-vel osztva pontosan 2 maradékosztályba tartoznak.

Két konstrukciót is mutatunk:

- Legyen a polinom x^{p-1}. A 0 helyen ez 0-t vesz fel, az 1,...$p-1$ helyeken pedig p-vel osztva 1-et ad maradékul, lásd: kis-Fermat tétel.
- A feladat bizonyításában szereplő f polinom pontosan megfelel a követelményeknek.

A továbbiakban néhány speciális eset megoldását taglaljuk, valamint bemutatunk egy programot, amely az első néhány prímszámra és az összes értelmes k-ra megmondja, hogy a becslés éles-e, vagy se. A speciális esetekre kimondott állításoknak és a program helyességének a bizonyításához absztrakt algebrai fogalmakra és tételekre van szükség.

6. Absztrakt algebrai fogalmak és tételek [1]

Definíció: A G nem üres halmaz csoport, ha értelmezett rajta egy kétváltozós * művelet úgy, hogy

- a * művelet asszociatív, azaz minden $g, h, k \in G$ esetén $(g * h) * k = g * (h * k)$
- létezik $e \in G$ kétoldali neutrális elem, melyre $e * g = g * e$ teljesül minden $g \in G$-re
- minden $g \in G$-nek van kétoldali g^{-1} inverze, melyre $g * g^{-1} = g^{-1} * g = e$

Definíció: A csoport rendje a csoport elemszáma.

Definíció: Egy g csoportelem rendje a különböző hatványainak száma, jele $o(g)$.

4. Tétel: $o(g^k) = \frac{o(g)}{(o(g),k)}$, ahol $(o(g), k)$ a k és $o(g)$ legnagyobb közös osztóját jelöli.

Definíció: A T nem üres halmaz test, ha értelmezhető benne az összeadás és a szorzás úgy, hogy

- Az összeadás asszociatív
- Az összeadás kommutatív
- Van összeadásra nézve nullelem.
- Minden elemnek van ellentettje
- A szorzás asszociatív
- A szorzás kommutatív
- Van a szorzásra nézve egységelem
- Minden nem 0 elemnek van inverze
- Tetszőleges $x, y, z \in T$ esetén igaz a disztributivitás: $(x + y)z = xz + yz$ és $z(x + y) = zx + zy$.

Definíció: $\mathbf{Z}_n := \{0, 1, ...n - 1\}$

5. Tétel: Legyen p prím, ekkor $\mathbf{Z}_p$ test.

Definíció: G ciklikus csoport, ha az egyik elemének hatványaiból áll. Ez az elem G generátora.

1. Állítás: Egy test invertálható elemei a szorzásra nézve csoportot alkotnak, ezt nevezzük a test multiplikatív csoportjának.
Ekkor $\mathbf{Z}_p$ nem 0 elemei alkotják $\mathbf{Z}_p$ multiplikatív csoportját.

Definíció: A g szám primitív gyök modulo m, ha hatványai kiadják az összes redukált maradékosztályt mod m.

6. Tétel: Pontosan akkor létezik primitív gyök mod m, ha m = 1, 2, 4, vagy egy páratlan prímhatvány, vagy annak kétszerese.

Következmény: $\mathbf{Z}_p$ multiplikatív csoportjában van primitív gyök modulo p. Az összes redukált maradékosztály modulo p viszont reprezentálható $\mathbf{Z}_p$ multiplikatív csoportjával, azaz $\mathbf{Z}_p$ multiplikatív csoportja ciklikus.

Definíció: Legyen G csoport a $*$ műveletre, és H csoport a $\bullet$ műveletre. Az $F : G \to H$ leképezés csoporthomomorfizmus, ha művelettartó: $F(a*b) = F(a) \bullet F(b)$ minden $a, b \in G$-re.

Definíció: Ha φ: $G \to H$ egy csoporthomomorfizmus, akkor legyen $\mathrm{Im}(\varphi) = \{\varphi(a); a \in G\} \subseteq H$ a φ képe.

Definíció: Ha φ: $G \to H$ egy csoporthomomorfizmus, akkor legyen $\mathrm{Ker}(\varphi) = \{a \in G : \varphi(a) = 1_H\} \subseteq G$ a φ magja (itt 1_H a H csoport egységeleme). Nyilván $\mathrm{Ker}(\varphi)$ részcsoport G-ben, és φ akkor és csak akkor injektív, ha $\mathrm{Ker}(\varphi) = \{1_G\}$

2. Állítás: A $\mathbf{Z}_p$ multiplikatív csoportjában az $x \to x^n$ homomorfizmus.

3. Állítás: Egy G csoport magja részcsoport G-ben.

4. Állítás: Legyen G véges ciklikus csoport, d osztója G rendjének. Ekkor G-nek pontosan d olyan g eleme van, amelyre $g^d = 1$.

7. Tétel: $|Im(\varphi)| = |G|/|Ker(\varphi)|$.

5. Állítás: Ha $a \neq 0$, akkor az $x \to ax$ bijekció $\mathbf{Z}_p$-ben.

6. Állítás: $f(x) = x^d + ux^{d-1} + ...$ és $f(x + c)$ értékkészlete ugyanakkora. Ebben x^{d-1} együtthatója $cd + u$ a binomiális tétel miatt.

7. Egész eset

8. Tétel: Ha a $(p - 1)/(k - 1)$ egész (jelöljük n-nel), akkor az x^n polinom teljesíti a feltételeket, vagyis a becslés éles.

Bizonyítás: Az egész kérdést elég $\mathbf{Z}_p$-ben vizsgálni, és át lehet fogalmazni úgy, hogy a polinom értékkészletének elemszámáról kell belátni, hogy pontosan k. Az 6. tétel következménye szerint a $\mathbf{Z}_p$ multiplikatív csoportja ciklikus, a rendje $p - 1$. Tekintsük az $x \to x^n$ leképezést, n a definíciója miatt osztója $p - 1$-nek. A 4. állítást felhasználva láthatjuk, hogy a leképezés magjának elemszáma n. Az 7. Tétel szerint a leképezés képének az elemszáma $(p - 1)/n = k - 1$. A 0 lesz a k-adik érték, amit a polinom felvesz az $x = 0$ helyen. Ezzel a tételt bebizonyítottuk.

8. A program

A következőkben egy **C++** nyelven írt programot mutatunk, amely kiszámolja minden p-re és k-ra, hogy melyik a legalacsonyabb fokú polinom $\mathbf{Z}_p$ fölött, amely értékkészlete k elemű. Az eredeti OKTV feladat megoldásából már kiderült, hogy ez legalább $(p - 1)/(k - 1)$-ed

fokú. Ezen kívül beláttuk, hogy a becslés éles, ha $k = 2$, valamint ha $(p-1)/(k-1)$ egész. Ezekkel az esetekkel a program nem is foglalkozik.

Egyszerűsítések: Az összes polinom végignézése nyilvánvalóan nem lehetséges, ezt csak egy adott prímszámig nézzük, esetünkben 29-ig. Még ez is nagyon sokáig tartana, ezért a polinomokat legfeljebb 7-ed fokig nézzük. Egy hetedfokú polinomnak 8 együtthatója van, azonban nekünk elég mindössze 5-öt legenerálni.

- A konstans tag választható 0-nak, a maradékosztályok számát nem befolyásolja.
- Az 5. állítás miatt a főegyüttható választható 1-nek.
- A 6. állítás alapján $c := -u/d$ választással az $n-1$-ed fokú tag együtthatója 0.

A kód:

```
#include <iostream>
#include <set>
#include <cmath>
using namespace std;
bool polinom (int fok, int p, int k)
{
    cout<<"p = "<<p<<"; k = "<<k<<"; fokszam: "<<fok<<endl;
    if(fok>7)
    {
        cout<<"erre mar nem fut le"<<endl;
        return false;
    }
    int egyutthato[fok-1];
    for (int i=0; i<fok-1; i++)
    {
        egyutthato[i]=0;
    }
    int h=fok-2;
    while (egyutthato[0]==0)
    {
        if (egyutthato[h]<p-1)
        {
            egyutthato[h]++;
        }
        else
        {
            while(egyutthato[h]==p-1)
            {
                h--;
                egyutthato[h+1]=0;
            }
            egyutthato[h]++;
        }
```

```
34
35          h=fok-2;
36          int x=0;
37          bool tulcsordult=false;
38          set<int> maradekosztaly;
39          while(tulcsordult==false&&x<p)
40          {
41              //Horner
42              int fuggvenyertek=x;
43              for (int i=fok-2; i>0; i--)
44              {
45                  fuggvenyertek = (x * fuggvenyertek + egyutthato[fok-i-1])%p;
46              }
47              fuggvenyertek=(fuggvenyertek * x)%p;
48
49              maradekosztaly.insert(fuggvenyertek);
50
51              if (maradekosztaly.size()>k)
52              {
53                  tulcsordult=true;
54              }
55              x++;
56          }
57          if (maradekosztaly.size()==k)
58          {
59
60              for (int j=1; j<fok-1; j++)
61              {
62                  cout<<egyutthato[j]<<" ";
63              }
64              cout<<endl;
65              return true;
66          }
67      }
68      cout<<fok<<". fokut nem talalt"<<endl;
69      polinom(fok+1,p,k);
70
71  }
```

```
72
73  bool isPrime(int n)
74  {
75      int i=0;
76      double x = sqrt((double) n);
77      for(i=2; i<=x; ++i)
78      {
79          if(n%i==0)
80          {
81              return false;
82          }
83      }
84      return true;
85  }
86  int main()
87  {
88      for (int p=3; p<30; p++)
89      {
90          for (int k=3; k<p; k++)
91          {
92              if(isPrime(p)==true && (p-1)%(k-1)>0)
93              {
94
95                  int fok=(p-1)/(k-1)+1;
96                  if(fok>1)
97                  {
98                      polinom(fok,p,k);
99                  }
100
101             }
102         }
103     }
104     return 0;
105 }
106
```

9. A program elemzése

A programban a **main** függvényen kívül két további függvény található, a **polinom(fok,p,k)**, és az **isPrime(n)**. Az utóbbi egyszerűbb, azt tárgyaljuk először. A függvény **int** típusú, változót, vagyis egész számot vár bemenetként, a kimenet pedig **igaz** vagy **hamis**. A visszatérési érték akkor igaz, ha a bemenetként kapott n prím, és hamis, ha nem. A módszer pedig mindössze annyi, hogy megvizsgáljuk az összes 2 és $\sqrt{n}$ közötti számot, hogy osztja-e n-t, és ha bármelyikről kiderül, hogy osztó, akkor tudjuk, hogy n nem prím, különben igen.

A **polinom(fok,p,k)** függvény három egész típusú változót vár bemenetként, **igaz** vagy **hamis** értéket ad kimenetként. Amennyiben létezik olyan f **fok** fokú polinom $\mathbf{Z}_p$ fölött,

mely értékkészlete k elemű, akkor a visszatérési érték igaz, és meg is ad egy ilyen polinomot. Az **Egyszerűsítések** fejezetben leírtak miatt mindig legfeljebb fokszám-2 együtthatót kell legenerálni.

Ennek módja a következő:
Létrehozunk egy **fok-1** elemű tömböt, úgy tekintünk rá, mint egy p számrendszerben lévő, legfeljebb **fok-1** számjegyű **M** számra. Minden lépésben **M**-hez hozzáadunk 1-et, egészen addig, amíg a legnagyobb helyiértéken lévő számjegy 0-ről 1-esre nem változik. Azaz lényegében **M** csak **fok-2** számjegyű, az utolsó helyiérték mindössze technikai megfontolásból kell. Amíg csak az a kérdés, hogy van-e ilyen polinom, addig természetesen nem kell minden együtthatót legenerálni, elég addig, amíg nem találtunk megfelelő polinomot.

Most azt tárgyaljuk, hogy adott együtthatók mellett hogyan kell eldönteni, hogy az ezekből álló polinom értékkészlete mekkora. Első lépésként be kell helyettesíteni $\mathbf{Z}_p$ összes elemét a polinomba, kivéve, ha már korábban túlcsordul, azaz egy $p-1$-nél kisebb x behelyettesítése után az értékkészlet k-nál nagyobb elemszámú. A behelyettesítés Horner elrendezéssel történik, hamarosan azt is tárgyaljuk, hogy pontosan hogyan, illetve miért pont így. A függvényértékeket mindig modulo p vizsgáljuk. Egy **halmaz (set)** adatszerkezetben tároljuk a őket, és minden lépés után megvizsgáljuk, hogy túlcsordult-e, azaz a halmaz mérete meghaladja-e k-t. Ha minden $x \in \mathbf{Z}_p$ behelyettesítése után nem csordult túl, akkor megvizsgáljuk, hogy pontosan k-e a halmaz mérete. Ha igen, akkor készen vagyunk, hiszen találtunk egy megfelelő polinomot, ezt ki is írjuk. Ha nem, akkor tovább kell generálni az együtthatókat.

A program valójában nem csak arra keresi a választ, hogy létezik-e **fok** fokszámú megfelelő polinom, hanem azt is, hogy melyik az a legkisebb fokszám, amire van találat. Ezért, ha a **fok** fokszám esetén nincs találat, akkor ezt kiírjuk, majd a fokszám növelésével újra lefuttatjuk, és ezt addig csináljuk amíg nincs találat, vagy nem kéne olyan fokszámot vizsgálni, amely csak nagyon hosszú idő alatt futna le.

A **main** függvényben pedig minden p pozitív egészről, ami nagyobb, mint 3, és kisebb, mint (pl.) 30, és minden ennél nem nagyobb k-ról először eldöntöm, hogy p prím-e, illetve, hogy $(p-1)/(k-1)$ egész-e. Ha p prím, $(p-1)/(k-1)$ pedig nem egész (lásd 8-as tétel), akkor meghívom a **polinom(fok,p,k)** függvényt.

Megjegyzés: a 8-as tételt először a program eredményei alapján lehetett megsejteni, de mivel sikerült bebizonyítani, ezért azokkal az esetekkel tovább nem foglalkozunk.

10. A program eredménye:

p	k	fok	$\lceil \frac{p-1}{k-1} \rceil$	f
5	4	4	2	$x^4 + x$
7	5	3	2	$x^3 + x$
7	6	6	2	$x^6 + x$
11	4	4	4	$x^4 + 2x^2$
11	5	4	3	$x^4 + x^2$
11	7	3	2	$x^3 + x$
11	8	4	2	$x^4 + x^2 + x$
11	9	5	2	$x^5 + x$
11	10	> 7	2	
13	6	4	3	$x^4 + 2x^2$
13	8	4	2	$x^4 + x^2 + x$
13	9	3	2	$x^3 + x$
13	10	4	2	$x^4 + 2x$
13	11	6	2	$x^6 + 2x^3 + 2x^2 + 12x$
13	12	> 7	2	
17	4	> 7	6	
17	6	6	4	$x^6 + x^4 + x^2$
17	7	4	3	$x^4 + x^2$
17	8	5	3	$x^5 + x^2 + 13x$
17	10	4	2	$x^4 + x^2 + 5x$
17	11	3	2	$x^3 + x$
17	12	4	2	$x^4 + x^2 + 4x$
17	13	4	2	$x^4 + x^2 + 8x$
17	14	6	2	$x^6 + x^4 + 2x^2 + 3x$
17	15	> 7	2	
17	16	> 7	2	
19	5	6	5	$x^6 + x^3$
19	6	6	4	$x^6 + 4x^3$
19	8	4	3	$x^4 + x^2$
19	9	5	3	$x^5 + x^3 + 8x$
19	11	4	2	$x^4 + x^2 + 3x$
19	12	4	2	$x^4 + x^2 + 4x$
19	13	3	2	$x^3 + x$
19	14	4	2	$x^4 + x^3 + x^2 + 2x$
19	15	5	2	$x^5 + x^4 + x^3 + 2x$
19	16	6	2	$x^6 + x^5 + x^3 + 5x^2 + 13$x

19	17	> 7	2	
19	18	> 7	2	
23	4	> 7	8	
23	5	> 7	6	
23	6	> 7	5	
23	7	> 7	4	
23	8	6	4	$x^6 + x^4 + 6x^2$
23	9	4	3	$x^4 + x^2$
23	10	5	3	$x^5 + 5x^3 + 7x^2 + 9x$
23	11	5	3	$x^5 + x^3 + 2x$
23	13	4	2	$x^4 + 5x^2 + 3x$
23	14	4	2	$x^4 + x^2 + 3x$
23	15	3	2	$x^3 + x$
23	16	4	2	$x^4 + x^2 + x$
23	17	4	2	$x^4 + x^2 + 10x$
23	18	5	2	$x^5 + x^2 + 19x$
23	19	6	2	$x^6 + x^4 + 10x^2 + 8x$
23	20	> 7	2	
23	21	> 7	2	
23	22	> 7	2	
29	4	> 7	10	
29	6	> 7	6	
29	7	> 7	5	
29	9	6	4	$x^6 + x^4 + 21x^2$
29	10	6	4	$x^6 + x^4 + 9x^2$
29	11	4	3	$x^4 + x^2$
29	12	4	3	$x^4 + 2x^2$
29	13	5	3	$x^5 + 2x^3 + 12x$
29	14	5	3	$x^5 + x^3 + 6x^2 + 3x$
29	16	5	2	$x^5 + x^2 + 6x$
29	17	4	2	$x^4 + x^2 + x$
29	18	4	2	$x^4 + x^2 + 3x$
29	19	3	2	$x^3 + x$
29	20	4	2	$x^4 + x^2 + 4x$
29	21	5	2	$x^5 + 4x$
29	22	5	2	$x^5 + x^2 + 23x$
29	23	5	2	$x^5 + x^3 + 5x$
29	24	6	2	$x^6 + x^4 + 11x^2 + 10x$
29	25	7	2	$x^7 + 2x^3$
29	26	> 7	2	
29	27	> 7	2	
29	28	> 7	2	

11. Polinomok száma

Egy másik program, melynek működését nem részletezzük, azt a kérdést feszegette, hogy adott p és adott fokszám mellett az összes lehetséges k-ra végignézi, hogy hány polinom van, melynek az értékkészlete k elemű. A polinomokról itt is feltételezzük, hogy a főegyüttható 1, az $n-1$-ed fokú tag együtthatója és a konstans tag pedig 0. Ennek az eredményei a következők:

$p = 5$

fokszám	k	polinomok száma
2		
	1	0
	2	0
	3	1
	4	0
	5	0
3		
	1	0
	2	0
	3	4
	4	0
	5	1
4		
	1	0
	2	3
	3	10
	4	12
	5	1

$p = 7$

fokszám	k	polinomok száma
2		
	1	0
	2	0
	3	0
	4	1
	5	0
	6	0
	7	0

fokszám	k	polinomok száma
3		
	1	0
	2	0
	3	1
	4	0
	5	6
	6	0
	7	0
4		
	1	0
	2	0
	3	6
	4	11
	5	30
	6	0
	7	2
5		
	1	0
	2	0
	3	36
	4	88
	5	204
	6	0
	7	22048
6		
	1	0
	2	9
	3	172
	4	900
	5	960
	6	360
	7	0

$p = 11$

fokszám	k	polinomok száma
2		
	1	0
	2	0
	3	0
	4	0
	5	0
	6	1

fokszám	k	polinomok száma
	7	0
	8	0
	9	0
	10	0
	11	0
3		
	1	0
	2	0
	3	0
	4	0
	5	0
	6	0
	7	10
	8	0
	9	0
	10	0
	11	1
4		
	1	0
	2	0
	3	0
	4	5
	5	5
	6	11
	7	50
	8	50
	9	0
	10	0
	11	1
5		
	1	0
	2	0
	3	1
	4	0
	5	75
	6	260
	7	520
	8	340
	9	135
	10	0
	11	1

fokszám	k	polinomok száma
6		
	1	0
	2	0
	3	5
	4	60
	5	710
	6	2982
	7	5670
	8	3760
	9	1430
	10	0
	11	25
7		
	1	0
	2	0
	3	10
	4	660
	5	7835
	6	32806
	7	62710
	8	40730
	9	16075
	10	0
	11	250
8		
	1	0
	2	0
	3	225
	4	6495
	5	89215
	6	355692
	7	692300
	8	450160
	9	174720
	10	0
	11	3004
9		
	1	0
	2	0
	3	2350
	4	72280
	5	978800
	6	3915936
	7	7615220

fokszám	k	polinomok száma
	8	4948160
	9	1924440
	10	0
	11	32989
10		
	1	0
	2	93
	3	20728
	4	874500
	5	10228080
	6	45034920
	7	79576560
	8	59875200
	9	16934400
	10	1814400
	11	32989

$p = 13$

fokszám	k	polinomok száma
2		
	1	0
	2	0
	3	0
	4	0
	5	0
	6	0
	7	1
	8	0
	9	0
	10	0
	11	0
	12	0
	13	0
3		
	1	0
	2	0
	3	0
	4	0
	5	1
	6	0
	7	0
	8	0
	9	12

fokszám	k	polinomok száma
	10	0
	11	0
	12	0
	13	0
4		
	1	0
	2	0
	3	0
	4	1
	5	6
	6	6
	7	4
	8	72
	9	60
	10	20
	11	0
	12	0
	13	0
5		
	1	0
	2	0
	3	0
	4	0
	5	9
	6	60
	7	406
	8	684
	9	675
	10	344
	11	0
	12	0
	13	19
6		
	1	0
	2	0
	3	1
	4	20
	5	124
	6	1086
	7	4530
	8	9240
	9	9288
	10	3312
	11	960

fokszám	k	polinomok száma
	12	0
	13	19
7		
	1	0
	2	0
	3	0
	4	44
	5	1362
	6	14136
	7	61586
	8	117708
	9	118902
	10	46316
	11	11124
	12	0
	13	134
8		
	1	0
	2	0
	3	18
	4	801
	5	18204
	6	182070
	7	797188
	8	1546596
	9	1523160
	10	615408
	11	141984
	12	0
	13	1514
9		
	1	0
	2	0
	3	120
	4	8364
	5	246910
	6	2346684
	7	10379604
	8	20108892
	9	19792095
	10	7996584
	11	1852524
	12	0
	13	18254

fokszám	k	polinomok száma
10		
	1	0
	2	0
	3	1422
	4	110674
	5	3198510
	6	30550854
	7	134850945
	8	261479976
	9	257299968
	10	103929056
	11	24091296
	12	0
	13	236274

p=17

fokszám	k	polinomok száma
2		
	1	0
	2	0
	3	0
	4	0
	5	0
	6	0
	7	0
	8	0
	9	1
	10	0
	11	0
	12	0
	13	0
	14	0
	15	0
	16	0
	17	0
3		
	1	0
	2	0
	3	0
	4	0
	5	0
	6	0
	7	0
	8	0

fokszám	k	polinomok száma
	9	0
	10	0
	11	16
	12	0
	13	0
	14	0
	15	0
	16	0
	17	1
4		
	1	0
	2	0
	3	0
	4	0
	5	1
	6	0
	7	16
	8	0
	9	0
	10	64
	11	128
	12	64
	13	16
	14	0
	15	0
	16	0
	17	1
5		
	1	0
	2	0
	3	0
	4	0
	5	0
	6	0
	7	16
	8	128
	9	464
	10	1104
	11	1704
	12	864
	13	616
	14	0
	15	0
	16	0
	17	18

fokszám	k	polinomok száma
6		
	1	0
	2	0
	3	0
	4	0
	5	4
	6	80
	7	296
	8	1696
	9	8453
	10	20016
	11	25760
	12	18448
	13	6656
	14	2112
	15	0
	16	0
	17	18
7		
	1	0
	2	0
	3	0
	4	0
	5	40
	6	240
	7	4272
	8	33712
	9	147544
	10	327808
	11	439768
	12	315520
	13	119176
	14	31568
	15	0
	16	0
	17	227
8		
	1	0
	2	0
	3	1
	4	40
	5	328
	6	5552
	7	73064

fokszám	k	polinomok száma
	8	581184
	9	2449608
	10	5688768
	11	7381120
	12	5354368
	13	2136496
	14	419552
	15	47488
	16	0
	17	227
9		
	1	0
	2	0
	3	0
	4	0
	5	2160
	6	76064
	7	1254032
	8	9907280
	9	41759634
	10	96424640
	11	125707416
	12	90872928
	13	36415272
	14	7133200
	15	783024
	16	0
	17	3250

$p = 19$

fokszám	k	polinomok száma
2		
	1	0
	2	0
	3	0
	4	0
	5	0
	6	0
	7	0
	8	0
	9	0
	10	1
	11	0

fokszám	k	polinomok száma
	12	0
	13	0
	14	0
	15	0
	16	0
	17	0
	18	0
	19	0
3		
	1	0
	2	0
	3	0
	4	0
	5	0
	6	0
	7	1
	8	0
	9	0
	10	0
	11	0
	12	0
	13	18
	14	0
	15	0
	16	0
	17	0
	18	0
	19	0
4		
	1	0
	2	0
	3	0
	4	0
	5	0
	6	0
	7	9
	8	9
	9	0
	10	7
	11	36
	12	180
	13	66
	14	54
	15	0

fokszám	k	polinomok száma
	16	0
	17	0
	18	0
	19	0
5		
	1	0
	2	0
	3	0
	4	0
	5	0
	6	0
	7	0
	8	18
	9	207
	10	456
	11	1278
	12	1980
	13	1866
	14	720
	15	333
	16	0
	17	0
	18	0
	19	1
6		
	1	0
	2	0
	3	0
	4	1
	5	6
	6	33
	7	105
	8	324
	9	2232
	10	10404
	11	26442
	12	36522
	13	32796
	14	16524
	15	3834
	16	1098
	17	0
	18	0
	19	1

fokszám	k	polinomok száma
7		
	1	0
	2	0
	3	0
	4	0
	5	0
	6	18
	7	495
	8	6048
	9	48825
	10	199350
	11	490113
	12	708642
	13	608712
	14	313470
	15	84240
	16	16074
	17	0
	18	0
	19	113
8		
	1	0
	2	0
	3	0
	4	9
	5	72
	6	882
	7	10287
	8	122301
	9	905526
	10	3807232
	11	9295578
	12	13481496
	13	11555460
	14	5970330
	15	1580004
	16	315840
	17	0
	18	0
	19	977
9		
	1	0
	2	0
	3	1

fokszám	k	polinomok száma
	4	12
	5	288
	6	6138
	7	172770
	8	2323152
	9	17392374
	10	72254628
	11	176554242
	12	255501720
	13	221178906
	14	111449970
	15	31852638
	16	4804524
	17	380376
	18	0
	19	977

Sejtés: ha a fokszám rögzített és p-vel tartunk a végtelenhez, akkor a polinomok száma a normális eloszláshoz konvergál.

12. Horner vagy behelyettesítés?

Egy adott polinomba egy adott x értéket több módon is be lehet helyettesíteni, és amikor ekkora számokkal kell ennyire sokat számolni, akkor a futásidő nem mindegy, hogy mennyi. Éppen ezért megvizsgálom, hogy a polinomok helyettesítési értékét Horner elrendezéssel vagy egyszerűen behelyettesítéssel gyorsabb kiszámolni.

Behelyettesítés algoritmusa:

```
behelyettesítés(x, n, hatvány[1...n], együttható[1...n-1], p):
hatvány[1] := x
i = 2...n
    hatvány[i] := (hatvány[i-1]*x) mod p
függvényérték := 0
i=1...(n-1)
    függvényérték := (függvényérték+hatvány[i]*együttható[i]) mod p
függvényérték := (függvényérték+hatvány[n]) mod p
return függvényérték
```

Műveletigény: Az első ciklusban $n-1$ szorzás és $n-1$ modulo, a másodikban $n-1$ összeadás, $n-1$ szorzás és $n-1$ modulo. Összesen nagyjából $5n$ művelet. Megjegyzés: a modulo p-re azért van szükség, mert nagy számok esetén a változók gyorsan túlcsordulhatnak. Ha tudjuk, hogy nem merülhet fel ez a probléma, akkor modulo nélkül $3n$ műveletből meg lehet oldani. Köztes megoldás lehet, és talán ez is a legcélszerűbb, hogy csak az első ciklusban számolunk modulo, hiszen leginkább ott tud túlcsordulni, a második ciklusban minden alkalommal legfeljebb p^2-tel növelünk, összesen $n-1$-szer, ekkor a műveletigény nagyjából $4n$.

Tárhelyigény: az együttható tömb adott, a hatvány tömb foglal n helyet.

A Horner elrendezés algoritmusa:

```
Horner (x, n, együttható[1...n-1], p):
függvényérték := 1
i = (n-1)...1
    függvényérték :=(x * függvényérték + együttható[i])mod p
függvényérték := (függvényérték * x)mod p
return függvényérték
```

Műveletigény: A ciklusban van $n-1$ szorzás, $n-1$ összeadás és $n-1 modulo$. Összesen nagyjából $3n$ művelet, ugyanazért érdemes modulo p-vel számolni, mint az előző esetben.

Tárigény: az együttható tömb adott, az algoritmus több helyet nem foglal.

Így a mi esetünkben a Horner elrendezés egyértelműen hatékonyabb. Felmerül a kérdés, hogy vajon ez mindig így van-e. A dolgozat fő témájától kicsit elrugaszkodva egy másik esetben is összehasonlítom a két módszert.

Feladat: Tekintsük az összes n-ed fokú polinomot $\mathbf{Z}_p$ fölött. Ezekbe helyettesítsük be az összes $\mathbf{Z}_p$-beli x számot. Kérdés, hogy ekkor melyik algoritmus a hatékonyabb.

A Horner elrendezés elemzése: Az algoritmus hasonló, mint az előző esetben, a főegyüttható nem feltétlenül 1, a konstans tag nem feltétlenül 0, de a műveletigényt ez nagyságrendileg nem befolyásolja. Ezt futtatom minden minden $\mathbf{Z}_p$ feletti polinomra és $x \in \mathbf{Z}_p$-re. A főegyüttható nem 0, így összesen $(p-1)p^n$ ilyen polinom van. Megfelelő x-ből pedig p van, bár $x = 0$-ra értelmetlen lefuttatni, ezért inkább $p-1$.Így a műveletigény összesen $3n(p-1)^2p^n \approx 3np^{n+2}$.

A behelyettesítéses módszer elemzése: Ha az algoritmust a főegyütthatón és a konstans tagon kívül változtatás nélkül hagyom, és ezt futtatom minden minden $\mathbf{Z}_p$ feletti polinomra és $x \in \mathbf{Z}_p$-re, akkor $(p-1)^2 p^n 5n \approx 5np^{n+2}$ a műveletigény. De itt minden hatványt sokszor kiszámoltunk, ezen lehet javítani.

Az x hatványait számoljuk ki előre és tároljuk el! Vagyis az eredeti algoritmus első ciklusát csak p-1-szer kell futtatni. Ennek a műveletigénye nagyjából $2pn$.
A második ciklusban, ha nem modulo p számolok, akkor azon belül $2n$ művelet van, és $(p-1)^2 p^n$-szer fut le (a ciklus legvégén értelem szerűen kell még egy modulo p). Így összesítésben a műveletigény $2n(p-1) + 2n(p-1)^2 p^n \approx 2np^{n+2}$. Tehát ebben az esetben a behelyettesítés gyorsabb, mint a Horner elrendezés.

A behelyettesítéses módszer feladatra módosított algoritmusa:

```
x_hatvány(x, n, p):
hatvány[0] := 1
i = 1...n
    hatvány[i] := (hatvány[i-1]*x) mod p
```

Egy adott polinomra (együttható[0...n]):

```
függvényérték := 0
i=0... n
    függvényérték := (függvényérték+hatvány[i]*együttható[i])
return függvényérték mod p
```

Elmélet és valóság Az előző feladatban minden műveletet egységnek tekintve arra a következtetésre juthatunk, hogy a behelyettesítés másfélszer gyorsabb a Horner elrendezésnél. Az eredeti feladathoz írt program egy részét módosítva futtattam mindkét módszerrel és az alábbi eredményekre jutottam:

p	n	Horner(sec)	Behelyettesítés (sec)
5	6	0,188	0,105
5	7	0,218	0,135
5	8	0,904	0,402
5	9	3,572	1,860
7	6	0,405	0,242
7	7	2,529	1,272
7	8	20,218	9,823
7	9	169,254	75,323
11	6	13,044	6,306
11	7	162,407	70,180

Ebből megállapíthatjuk, hogy gyakorlatban inkább picivel több, mint kétszeres a különbség.

Az alábbi két fejezet a programok futtatása után, részben az eredmények hatására született, így az itteni eredmények már nincsenek a programba építve.

13. Hatványösszegek

9. Tétel[2]**:** Ha f foka n, akkor

$$n \geq \frac{p-1}{p-k}\,.$$

Bizonyítás: Használjuk fel az alábbi lemmát: tegyük fel, hogy $1 \leq m < p$ és $b_1, \ldots, b_m, c_1, \ldots, c_m \in \mathbf{Z_p}$ továbbá

$$b_1^j + \ldots + b_m^j = c_1^j + \ldots + c_m^j$$

minden $1 \leq j \leq m$ esetén. Ekkor a $b_1, \ldots, b_m$, illetve a $c_1, \ldots, c_m$ elemek halmaza multiplicitásokkal tekintve is megegyezik. Valóban, $m < p$ miatt a Newton-Girard-formulákból indukcióval következik, hogy a $b_1, ..., b_m$ és a $c_1, ..., c_m$ elemek elemi szimmetrikus polinomjai is megegyeznek, ezért ugyanannak a normált polinomnak a gyökeiről van szó.

Legyen f értékkészlete $a_1, \ldots, a_k$ az x_i multiplicitásokkal és $m = p-k$. Válasszuk a $b_1, \ldots, b_m$ elemeket az értékkészlet komplementerének, $c_1, \ldots, c_m$ pedig legyenek az a_i elemek, ahol a_i-t $x_i - 1$-szer soroljuk fel. Nyilván

$$0^j + 1^j + \ldots + (p-1)^j = x_1 a_1^j + \ldots + x_k a_k^j + b_1^j + \ldots + b_m^j - c_1^j - \ldots - c_m^j\,.$$

Ha $j < p-1$, akkor a bal oldal nulla, és ha $nj < p-1$ is teljesül, akkor a 3. tétel miatt $x_1 a_1^j + \ldots + x_k a_k^j = 0$, így $b_1^j + \ldots + b_m^j = c_1^j + \ldots + c_m^j$. A lemma szerint ez nem lehetséges mindegyik $1 \leq j \leq m$ értékre, hiszen a b_i-k halmaza diszjunkt a c_i-k halmazától. Ezért $nm \geq p-1$. Ezzel a 9. tételt beláttuk. Megjegyzés: Ez a tétel valamilyen formában várhatóan megjelenik Kömal feladatként.

14. p=4m+1 eset

10. Tétel: Legyen $p = 4m + 1, m > 1$ egész. Ekkor $x^{2m} + x^m$ értékkészlete 4 elemű.

Bizonyítás: Legyen g primitív gyök, és $i = g^m$. Nem véletlen a jelölés, i a 4. tétel miatt tényleg negyedrendű, mivel $o(g) = p - 1 = 4m, (4m, m) = m$. Legyen $a = x^m$. Ez 4 féle értéket vehet fel, mivel a hatványa i-nek (pl. $x = g^l$, ekkor $a = (g^l)^m = g^{ml} = i^l$. Vagyis a lehetséges értékek $\pm i, \pm 1$. Ahogyan $a^2 + a$ is négyféle lehet, már csak azt kell belátni, hogy tényleg mind különböző. Tegyük fel, hogy $a^2 + a = b^2 + b$, de $a \neq b$. Átrendezve $(a - b)(a + b + 1) = 0$. Ha $a \neq b$, akkor $a + b + 1 = 0$. Ez $\binom{n-1}{k-1} = 6$ eset: $i + 1, i - 1, i + (-i), 1 + (-1), 1 + (-i), -1 + (-i)$. Kérdés, hogy i mennyi, ha minden összeg -1. Az $i + (-i)$ esetben nincs megoldás, a többi esetben $i = 0$, ami nem lehet, vagy $i = \pm 2$. De $i^4 = 1$, vagyis p osztója a 16-1-nek, vagyis $p = 3$ vagy 5. Nyilván 3-ra nincs 4 elemű értékkészlet, 5-re meg $x^2 + x$ értékkészlete 0,1,2, azaz 3 elemű. Beláttuk, hogy minden más p prímre nem fordulhat elő, hogy $a \neq b$. Tehát tényleg mind különböző. Ezzel a 10. tételt bebizonyítottuk.

Hivatkozások

[1] Kiss Emil: Bevezetés az algebrába (2007)
[2] Károlyi Gyula

Ezen kívül felhasználásra került a 2019-es OKTV döntő egyik feladata és megoldása. A többi eredmény a témavezetővel közös.

Tartalom