TALLER PER "UNA BREU HISTÒRIA DELS NOMBRES PRIMERS: D'EUCLIDES A TATE"

FRANCESC FITÉ

ABSTRACT. El Sage és un programa de codi obert i accés gratuït que s'utilitza per fer càlculs matemàtics. L'objectiu bàsic d'aquest taller és aprendre algunes comandes de Sage i el seu objectiu final és programar el test de primeritat de Lucas-Lehmer. El test de Lucas-Lehmer ha estat utilitzat per trobar alguns dels primers més grans que es coneixen. Com a preparació, implementarem el test de Lucas i també plantejarem alguns problemes de caire més teòric per estimular la curiositat dels estudiants.

El major primer conegut. El nombre primer més gran que es coneix a data d'avui (octubre de 2022) és:

$$2^{82.589.933} - 1$$

i té 24.862.048 dígits. Ostenta el rècord des de 2018. Els nombres que són de la forma

$$M_n = 2^n - 1$$

s'anomenen nombres de Mersenne. Un primer de Mersenne és un nombre de Mersenne que, a més a més, és primer. Molts dels primers que al llarg de la història han ostentat el rècord de ser el primer més gran conegut fins al moment eren primers de Mersenne. Heus aquí alguns primers de Mersenne

$$2^2 - 1 = 5, \quad 2^3 - 1 = 7, \quad 2^5 - 1 = 31, \quad 2^7 - 1 = 127.$$

Problema. Observeu que els exponents en els primers de Mersenne anteriors són al seu torn nombres primers. És sempre cert que, si M_n és un primer de Mersenne, aleshores n ha de ser primer?

La successió de nombres de Lucas. Els dos primers nombres de Lucas són 1 i 3. Cada nombre següent a la successió de Lucas s'obté com la suma dels dos anteriors:

$$l_1 = 1, \quad l_2 = 3, \quad l_3 = 4, \quad l_4 = 7, \quad l_5 = 11, \quad l_6 = 18, \quad l_7 = 29, \quad ...$$

El següent codi del programa Sage permet calcular nombres de Lucas:

```
def Lucas(n):
    if n==1:
        return 1
    if n==2:
        return 3
    else:
        return Lucas(n-1)+Lucas(n-2)
```

Problema. Raoneu perquè aquest codi calcula els nombres de Lucas. Utilitzant aquest codi, obtingueu els valors l_{10} i l_{15}. Tot el que heu de fer és teclejar:

```
Lucas(10)
Lucas(15)
```

Una matriu és una taula de valors. Heus aquí un exemple de matriu:

$$A = \begin{pmatrix} 5 & 1 & 4 \\ 1 & 1 & 9 \\ 5 & 3 & 4 \end{pmatrix}$$

Podem construir la matriu anterior en Sage indicant cadascuna de les línies horitzontals de la següent manera:

```
A=matrix([[5,1,4],[1,1,9],[5,3,4]])
```

Problema. Teclegeu

```
A[0][0]
A[1][2]
```

Podeu predir el valor de `A[2][1]`? Si no n'heu après a l'institut encara, demaneu que us ensenyin a multiplicar matrius.

Ara veurem que podem obtenir la successió de Lucas multiplicant matrius. Observem que

$$\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} l_1 \\ l_2 \end{pmatrix} = \begin{pmatrix} l_2 \\ l_1 + l_2 \end{pmatrix} = \begin{pmatrix} l_2 \\ l_3 \end{pmatrix}$$

De manera semblant tenim que

$$\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^2 \cdot \begin{pmatrix} l_1 \\ l_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} l_2 \\ l_3 \end{pmatrix} = \begin{pmatrix} l_3 \\ l_2 + l_3 \end{pmatrix} = \begin{pmatrix} l_3 \\ l_4 \end{pmatrix}$$

Del raonament anterior, deduïm que una manera alternativa de calcular nombres de Lucas amb Sage és la següent:

```
def LucasMatrius(n):
    if n==1:
        return 1
    if n==2:
        return 3
    else:
        A=matrix([[0,1],[1,1]])
        v=matrix([[1],[3]])
        return (A^(n-2)*v)[1][0]
```

Problema. Quin dels dos codis creieu que és més ràpid? Comproveu-ho tot calculant:

```
Lucas(31)
LucasMatrius(31)
```

Eina. Per cancel·lar l'execució d'una comanda en Sage, premeu Ctrl+c.

El test de Lucas. El test de Lucas permet determinar que certs nombres compostos ho són sense la necessitat de trobar la seva factorització en producte de nombres primers. Més en concret diu el següent:

> Si n no divideix $l_n - 1$, aleshores n no és primer. Si n divideix $l_n - 1$, aleshores és força probable que n sigui primer, però no en podem estar del tot segurs.

El següent codi de Sage aplica el test de Lucas a un nombre n:

```
def LucasTest(n):
    if n<=2:
        return "Introdueix un nombre major que 2."
    else:
        R=IntegerModRing(n)
        A=matrix(R,[[0,1],[1,1]])
        v=matrix(R,[[1],[3]])
        valor=(A^(n-2)*v)[1][0]-1
        if valor==0:
            return "Sembla primer però podria ser compost."
        else:
            return "És compost."
```

Remarca. El menor nombre compost no detectat pel test de Lucas és 705.

Problema. Comproveu que 10201 i 17834235 són nombres compostos sense necessitat de factoritzar-los. En teniu prou amb teclejar:

```
LucasTest(10201)
LucasTest(17834235)
```

El test de Lucas-Lehmer. El test de Lucas-Lehmer permet determinar si un nombre de Mersenne és un primer de Mersenne o no. Per aplicar-lo, necessitarem introduir la successió dels nombres de Lucas-Lehmer

$$s_i = \begin{cases} 4 & \text{si } i = 1, \\ s_{i-1}^2 - 2 & \text{si } i > 1. \end{cases}$$

És a dir, $s_1 = 4$ i després cada nombre s'obté com el quadrat de l'anterior menys 2:

$$s_1 = 4, \quad s_2 = 14, \quad s_3 = 194, \quad s_4 = 37634, \quad \dots$$

Aleshores el test de Lucas-Lehmer diu el següent:

> Sigui p un primer senar. Si s_{p-1} és divisible per M_p, aleshores M_p és primer; i si s_{p-1} no és divisible per M_p, aleshores M_p no és primer.

Remarca. Observeu que a diferència del test de Lucas, el test de Lucas-Lehmer és capaç de garantir que un cert nombre és primer.

Remarca. Donat un primer p, el test de Lucas-Lehmer ens permet determinar si un cert nombre M_p, que és mooolt més gran que p, és primer o no. Això ens proporciona una eina per construir primers grans a partir d'altres primers.

El següent codi de Sage aplica el test de Lucas-Lehmer al primer p:

```
def LucasLehmerTest(p):
    Mp=2^p-1
    s=mod(4,Mp)
    for n in range(2,p):
        s=mod(s^2-2,Mp)
    if s==0:
        return "M_p és primer."
    else:
        return "M_p no és primer."
```

Utilitzant el codi anterior, demostreu que M_{127} és primer. Aquest primer va ser descobert per Lucas el 1876. Va romandre durant 75 anys com el primer més gran conegut i és el darrer rècord obtingut calculant únicament a mà. Només va ser superat el 1951, amb l'entrada a l'era dels ordinadors.

Utilitzant el codi anterior, demostreu que $M_{19.937}$ és primer. Aquest fet va ser provat per primera vegada el 1971 per B. Tuckerman, i $M_{19.937}$ va romandre durant 7 anys com el primer més gran conegut.

Al següent enllaç de la Wikipèdia, podeu trobar la llista de rècords que l'han anat succeint:

https://en.wikipedia.org/wiki/Largest_known_prime_number

Descarregar Sage. Sage és gratuït. Si us ha agradat utilitzar-lo, el podeu instal·lar al vostre ordinador de casa a través de l'enllaç:

https://www.sagemath.org/

Bibliografia. Podeu trobar demostracions del tests considerats en aquest taller al llibre

Prime numbers. A computational perspective.

dels autors R. Grandall and C. Pomerance. En concret, el test de Lucas es demostra a la secció 3.6 i el test de Lucas-Lehmer a la secció 4.2.1.

Departament de matemàtiques i informàtica, i Centre de recerca matemàtica, Universitat de Barcelona, Gran via de les Corts Catalanes 585, 08007 Barcelona, Catalunya
Email address: ffite@ub.edu
URL: http://www.ub.edu/nt/ffite/