Algebra och diskret matematik Projektuppgift 1.5 hp

MA2047 Algebra och diskret matematik

Rapportversion 1, inlämnad 2021 - 10 - 31

Grupp

- Daniel Bleckert
- Erik Knutsson
- Vincent Larewall
- Linus Frisk

Uppgifter

Instruktioner

- Varje deluppgift i projektet ska redovisas enligt nedan.
- Krav: Filen ska klara “Ctrl-A” + “Shift-return” utan att lämna något felmeddelande.
- Handledningar till Mathematica hittar du här: http://dixon.hh.se/mikael/links.shtml
- Svensk rättstavning kan väljas i
  Format->Option inspector->Formatting options->Text content options
  eller genom att köra:

```
In[119]:= CurrentValue[EvaluationNotebook[], DefaultNaturalLanguage] = "Swedish";
```

- Rapporten måste klara “Ctrl-A + shift-return” utan att lämna något felmeddelande!

1)

Problem

Använd Eulers sats för att beräkna 7^{82} mod 100 för hand. Du kan bestämma $\phi(n)$ för lämpligt n med hjälp av Mathematicafunktionen EulerPhi. Kontrollera ditt svar med PowerMod.

Lösning

Vi använder Eulers sats enligt projektbeskrivningen[1].
7^{82} mod(100)

$a^{\phi n} \equiv 1 \bmod(n) \Rightarrow a^{\phi 100} \equiv 1 \bmod(100) \Leftrightarrow 7^{40} \equiv 1 \bmod(100)$

$7^{40} \equiv 1 \bmod(100)$

$7^{82} = \left(7^{40}\right)^2 \cdot 7^2 \equiv \left(1^2\right) \cdot 7^2 = 49 \equiv 49 \bmod(100)$

Mathematicakod

In[120]:=
```
Remove["Global`*"]
Print["EulerPhi: ", EulerPhi[100]]
Print["PowerMod: ", PowerMod[7, 82, 100]]
```

```
EulerPhi: 40
PowerMod: 49
```

Slutsatser och resultat

7^{82} mod (100) = 49

2)

Problem

Beräkna 572^{29} mod 713 med hjälp av “kvadreringsmetoden” ovan.

Lösning

Vi använder kvadreringsmetoden enligt projektbeskrivningen[1].

$572^{29} \bmod (713)$

$29 = 16 + 8 + 4 + 1$

$572^2 = 327184 = 458 * 713 + 630 \equiv 630 \bmod(713)$

$572^4 = \left(572^2\right)^2 = 630^2 = 396900 = 556 * 713 + 472 \equiv 472 \bmod(713)$

$572^8 = \left(572^4\right)^2 = 472^2 = 222784 = 312 * 713 + 328 \equiv 328 \bmod(713)$

$572^{16} = \left(572^8\right)^2 = 328^2 = 107584 = 150 * 713 + 634 \equiv 634 \bmod(713)$

$572^{29} = 572^{16} * 572^8 * 572^4 * 572 \equiv 634 * 328 * 472 * 572 = 56\,143\,712 = 78742935 * 713 + 113 \equiv 113 \bmod(713)$

Mathematicakod

In[123]:=
```
572^2
```

Out[123]=
```
327 184
```

Slutsatser och resultat

$572^{29} \bmod (713) = 113$.

3)

Problem

Bestäm primtalen p och q om n = pq = 39247771 och ϕ(n) = 39233944.

Lösning

$$\begin{cases} p*q = 39\,247\,771 & \Leftrightarrow p = \frac{39\,247\,771}{q} \\ (p-1)*(q-1) = 39\,233\,944 & \Longleftrightarrow p = \frac{39\,233\,944}{q-1} + 1 \end{cases}$$

$\frac{39\,247\,771}{q} = \frac{39\,233\,944}{q-1} + 1 \Leftrightarrow$ q^2-13828q + 39247771 = 0 $\Leftrightarrow$ q = 6914 ± 2925

q1 = 3989 ⇒ p1 = 9839

q2 = 9839 ⇒ p2 = 3989

Mathematicakod

```
In[124]:= Solve[q^2 - 13 828 q + 39 247 771 == 0, q]
Out[124]= {{q → 3989}, {q → 9839}}
```

Slutsatser och resultat

q = 3989 och p = 9839 eller q = 9839 och p = 3989.

4)

Problem

För heltalen a och b gäller att b ≡ a (mod 91) och sgd(a, 91) = 1.

(a) Bestäm ett positivt tal k > 1 sådant att b^k ≡ a (mod 91).

(b) Bestäm a mod 91 om b = 53.

Lösning

a)

Vi använder Eulers sats enligt projektbeskrivningen[1].

$a^{\phi n}$ ≡ 1 mod(n) enligt Eulers sats.

$b^{\phi 91}$ * b ≡ 1 * b mod (91) ⇔ b^{72} * b ≡ 1 * b mod (91)

b ≡ a mod (91) ⇔ b^{72} * b ≡ a mod (91) ⇔ b^{73} ≡ a mod (91)

b)

a = 91x + 53 där x ∈ $\mathbb{Z}$ och SGD(a, 91) = 1. Vi väljer a = 144.

Om a = 144 och b = 53 så stämmer b ≡ a mod (91) och SGD(a, 91) = 1.

Mathematicakod

In[125]:=
```
EulerPhi[91]
GCD[144, 91]
```
Out[125]= 72

Out[126]= 1

Slutsatser och resultat

a) k = 73 är en lösning.

b) a = 144 är en lösning.

5)

Problem

Bestäm några olika värden på e om p = 19 och q = 13. Välj därefter ett av värdena på e, bestäm d och kryptera meddelandet M = 10. Kontrollera dina beräkningar genom att avkryptera det krypterade meddelandet. Vilka meddelanden (tal) M kan krypteras med ovanstående nycklar?

Lösning

Vi använder information från projektbeskrivningen[1].

Vi kan välja e = 31 eller e = 71 eller e = 197. Vi väljer e = 31 och krypterar meddelandet M = 10 genom att ta 10^31 mod(p*q) = 10^31 mod(19*13).

K = 10^31 mod(247) = 127.

Avkryptering:

vi tar fram värdet på d som är den multiplikativa inversen till e alltså e*d ≡ 1 mod (ϕn) ⇔ 31*d ≡ 1 mod (216).

d = 7. A = K^d mod (n) ⇔ A = 127^7 mod(247) = 10.

M < 247. Om tex M = 247 så kommer vi inte få tillbaka samma meddelande som vi krypterade vid avkryptering.

K = 247^31 mod (247) = 0. A = K^7 mod (247) = 0. M ≠ A, detta betyder att vi inte kan kryptera meddelandet 247.

Mathematicakod

In[127]:=
```
19 * 13;
EulerPhi[247];
RandomPrime[{1, 216}];
PowerMod[10, 31, 247];
ModularInverse[31, 216]
```
Out[131]= 7

Slutsatser och resultat

Alla $M < 247$ där $M \in \mathbb{N}$ kan krypteras med nycklarna.

6)

Problem

Skriv ett program i Mathematica som krypterar och avkrypterar ett godtyckligt textmeddelande. Använd Unicode för tecknen då meddelandet transformeras till ett heltal och kryptera/avkryptera ett tecken i taget. Indata till programmet ska vara textmeddelandet, samt antalet siffror i primtalen q och p. Dessa ska sedan genereras slumpmässigt.

Lösning

Se Mathematicakod...

Mathematicakod

```
In[132]:= Remove["Global`*"]
m = (InputString["Message:"]);

In[134]:= a = (Input["Amount of digits in p:"]);
(*Skriv hur många siffror det ska finnas i p*)

In[135]:= x1 = 0;
y1 = {};

In[137]:= For[i = 1, i ≤ a, x1 = 10^(i-1);
 AppendTo[y1, 9];
 i++] (*Skapa minsta- och största
  värde för primtalet p*)


p = RandomPrime[10^(a-1), 10^a - 1]


In[138]:= y1 = FromDigits[y1];(*Gör om y1 till ett heltal*)

In[139]:= p = RandomPrime[{x1, y1}];(* Välj primtalet p i intervallet x1 till y1*)

In[140]:= b = (Input["Amount of digits in q:"]);
(*Skriv hur många siffror det ska finnas i q*)

In[141]:= x2 = 0;
y2 = {};
```

```
In[143]:= For[j = 1, j ≤ b, x2 = 10^(j-1);
           AppendTo[y2, 9];
           j++] (*Skapa minsta- och största
             värde för primtalet q*)

In[144]:= y2 = FromDigits[y2];(*Gör om y2 till ett heltal*)

In[145]:= q = RandomPrime[{x2, y2}]; (*Välj primtalet q i intervallet x2 till y2*)

In[146]:= While[q == p, q = RandomPrime[{x2, y2}]]
          (*Medan q är lika med p, tilldela q ett nytt primtal*)

In[147]:= n = p * q;

In[148]:= EulerPhi[n];

In[149]:= e = RandomPrime[{1, EulerPhi[n]}]; (*Ta fram ett värde på "e"*)

In[150]:= M = ToCharacterCode[m]; (*Gör om meddelandet till tal*)

In[151]:= K = PowerMod[M, e, n]; (*Kryptera meddelandet*)

In[152]:= d = ModularInverse[e, EulerPhi[n]]; (*Räkna ut värdet på "d"*)

In[153]:= A = PowerMod[K, d, n]; (*Avkryptera meddelandet*)

In[154]:= message = FromCharacterCode[A]; (*Gör om talen till tecken*)

In[155]:= Print[message];

          Carpe diem
```

Referenser

Här anges vilka källor som utnyttjats och som refereras till i rapporten. Ex:

[1] Projektuppgift 1.5 hp,
http://dixon.hh.se/mikael/teaching/algdisk/project/projekt_ma2047ht21.pdf