现代密码学作业——第六讲

1、由n = 35,知p=7，q=5.所以Φ（35）=24，易知d = 5，因此M = 105 mod 35 = 5

2、

（1）C=59,57

（2）K=3 m=30

（3）K=2 m=19

3、　Miller-Rabin(n,t)

　　输入：一个大于3的奇整数n和一个大于等于1的安全参数t(用于确定测试轮数)。

　　输出：返回n是否是素数(概率意义上的，一般误判概率小于(1/2)80即可) 。

　　1、将n-1表示成2sr，(其 中 r是奇数)

　　2、 对i从1到 t 循环作下面的操作：

　　 2.1选择一个随机整数a(2≤a ≤n-2)

　　 2.2计算y ←ar mod n

　　 2.3如果y≠1并且y ≠n-1作下面的操作,否则转3：

　　 2.3.1 j←1;

2.3.2 当j≤s-1 并且y≠n-1循环作下面操作,否则跳2.3.3：{计算y ←y2 mod n;

　　 如果 y=1返回 合数 ；

　　 否则 j←j+1; }

　　 2.3.3如果y ≠n-1 则返回 合数 ；

3、返回素数。、

1. #include <iostream>
2. #include <stdlib.h>
3. #include <time.h>
4. #include <math.h>
5. **using** **namespace** std;
7. // 生成伪素数
8. **const** **int** MAX\_ROW = 50;
9. **size\_t** Pseudoprime()
10. {
11. **bool** ifprime = **false**;
12. **size\_t** a = 0;
13. **int** arr[MAX\_ROW];   //数组arr为{3，4，5，6...52}
14. **for** (**int** i = 0; i<MAX\_ROW; ++i)
15. {
16. arr[i] = i+3;
17. }
18. **while** (!ifprime)
19. {
20. srand((unsigned)time(0));
21. ifprime = **true**;
22. a = (rand()%10000)\*2+3; //生成一个范围在3到2003里的奇数
23. **for** (**int** j = 0; j<MAX\_ROW; ++j)
24. {
25. **if** (a%arr[j] == 0)
26. {
27. ifprime = **false**;
28. **break**;
29. }
30. }
31. }
32. **return** a;
33. }
35. **size\_t**  repeatMod(**size\_t** base, **size\_t** n, **size\_t** mod)//模重复平方算法求(b^n)%m
36. {
37. **size\_t** a = 1;
38. **while**(n)
39. {
40. **if**(n&1)
41. {
42. a = (a\*base)%mod;
43. }
44. base = (base\*base)%mod;
45. n = n>>1;
46. }
47. **return** a;
48. }
50. //Miller-Rabin素数检测
51. **bool** rabinmiller(**size\_t** n, **size\_t** k)
52. {
54. **int** s = 0;
55. **int** temp = n-1;
56. **while** ((temp & 0x1) == 0 && temp)
57. {
58. temp = temp>>1;
59. s++;
60. }   //将n-1表示为(2^s)\*t
61. **size\_t** t = temp;
63. **while**(k--)  //判断k轮误判概率不大于(1/4)^k
64. {
65. srand((unsigned)time(0));
66. **size\_t** b = rand()%(n-2)+2; //生成一个b(2≤a ≤n-2)
68. **size\_t** y = repeatMod(b,t,n);
69. **if** (y == 1 || y == (n-1))
70. **return** **true**;
71. **for**(**int** j = 1; j<=(s-1) && y != (n-1); ++j)
72. {
73. y = repeatMod(y,2,n);
74. **if** (y == 1)
75. **return** **false**;
76. }
77. **if** ( y != (n-1))
78. **return** **false**;
79. }
80. **return** **true**;
81. }

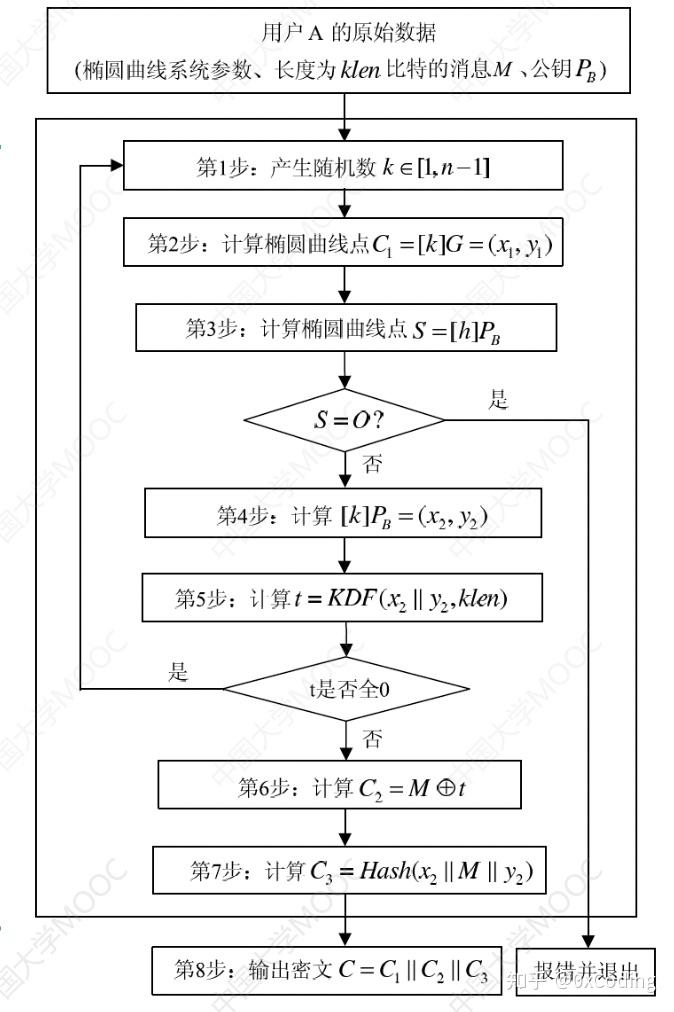
4、

**密钥产生**

设接收方为𝐵，𝐵的密钥取为{1,2,...,𝑛−1}的一个随机数𝑑𝐵，记为𝑑𝐵←𝑅{1,2,...,𝑛−1}，其中𝑛是基点𝐺的阶。

𝐵的公钥取为椭圆曲线上的点𝑃𝐵=𝑑𝐵𝐺，其中𝐺=𝐺(𝑥,𝑦)是基点。

**加密算法**



**解密算法**

