

运行说明

本程序只额外导入了random包，无其它依赖文件，可在IDE中直接运行。

实验步骤

1. 素数p的生成

```
30 def is_Prime(num):
31     # 小素数列表
32     smallPrime=[
33         2,3,5,7,11,13,17,19,23,29,31,37,41,43,47,53,59,61,67,71,73,79,83,89,97,101,103,107,109,
34         241,251,257,263,269,271,277,281,283,293,307,311,313,317,331,337,347,349,353,359,367,373
35         421,431,433,439,443,449,457,461,463,467,479,487,491,499,503,509,521,523,541,547,557,563
36         607,613,617,619,631,641,643,647,653,659,661,673,677,683,691,701,709,719,727,733,739,743
37         809,811,821,823,827,829,839,853,857,859,863,877,881,883,887,907,911,919,929,937,941,947
38     ]
39     if(num < 2):
40         return False
41
42     if num in smallPrime: # 在表中
43         return True
44
45     for prime in smallPrime:
46         if(num % prime == 0):
47             return False
48
49     return(Miller_Rabin(num))
50
```

使用Miller_Rabin函数来进行素性测试。

```
9 def Miller_Rabin(num): # 素性测试
10     safeTime = 128
11     s = num - 1
12     t = 0
13     while(s % 2 == 0):
14         s >>= 1
15         t = t + 1
16
17     for trials in range(safeTime):
18         random_a = random.randrange(2, num - 1)
19         v = pow(random_a, s, num) # random_a ** s % num
20         if v != 1:
21             i = 0
22             while v != (num - 1):
23                 if i == t - 1:
24                     return False
25                 else:
26                     i = i + 1
27                     v = (v ** 2) % num
28     return True
29
```

2. 求解素数p的本原根

其中注意从最小值开始遍历，当有多个本原根时，默认取最小值。

3. 基于蒙哥马利快速模幂运算计算通信双方的公钥 Y_a 和 Y_b ，验证是否符合条件

运行效果

194
 195
 196
 197
 198
 199
 200
 201
 202
 203
 204
 205
 206
 207
 208
 209
 210
 211
 212
 213
 214
 215
 216
 217
 218
 219
 220
 221
 222
 223
 224
 225
 226
 227
 228
 229
 230
 231
 232
 233
 234
 235
 236
 237
 238
 239
 240
 241
 242
 243
 244
 245
 246
 247
 248
 249
 250
 251
 252
 253
 254
 255
 256
 257
 258
 259
 260
 261
 262
 263
 264
 265
 266
 267
 268
 269
 270
 271
 272
 273
 274
 275
 276
 277
 278
 279
 280
 281
 282
 283
 284
 285
 286
 287
 288
 289
 290
 291
 292
 293
 294
 295
 296
 297
 298
 299
 300
 301
 302
 303
 304
 305
 306
 307
 308
 309
 310
 311
 312
 313
 314
 315
 316
 317
 318
 319
 320
 321
 322
 323
 324
 325
 326
 327
 328
 329
 330
 331
 332
 333
 334
 335
 336
 337
 338
 339
 340
 341
 342
 343
 344
 345
 346
 347
 348
 349
 350
 351
 352
 353
 354
 355
 356
 357
 358
 359
 360
 361
 362
 363
 364
 365
 366
 367
 368
 369
 370
 371
 372
 373
 374
 375
 376
 377
 378
 379
 380
 381
 382
 383
 384
 385
 386
 387
 388
 389
 390
 391
 392
 393
 394
 395
 396
 397
 398
 399
 400
 401
 402
 403
 404
 405
 406
 407
 408
 409
 410
 411
 412
 413
 414
 415
 416
 417
 418
 419
 420
 421
 422
 423
 424
 425
 426
 427
 428
 429
 430
 431
 432
 433
 434
 435
 436
 437
 438
 439
 440
 441
 442
 443
 444
 445
 446
 447
 448
 449
 450
 451
 452
 453
 454
 455
 456
 457
 458
 459
 460
 461
 462
 463
 464
 465
 466
 467
 468
 469
 470
 471
 472
 473
 474
 475
 476
 477
 478
 479
 480
 481
 482
 483
 484
 485
 486
 487
 488
 489
 490
 491
 492
 493
 494
 495
 496
 497
 498
 499
 500
 501
 502
 503
 504
 505
 506
 507
 508
 509
 510
 511
 512
 513
 514
 515
 516
 517
 518
 519
 520
 521
 522
 523
 524
 525
 526
 527
 528
 529
 530
 531
 532
 533
 534
 535
 536
 537
 538
 539
 540
 541
 542
 543
 544
 545
 546
 547
 548
 549
 550
 551
 552
 553
 554
 555
 556
 557
 558
 559
 560
 561
 562
 563
 564
 565
 566
 567
 568
 569
 570
 571
 572
 573
 574
 575
 576
 577
 578
 579
 580
 581
 582
 583
 584
 585
 586
 587
 588
 589
 590
 591
 592
 593
 594
 595
 596
 597
 598
 599
 600
 601
 602
 603
 604
 605
 606
 607
 608
 609
 610
 611
 612
 613
 614
 615
 616
 617
 618
 619
 620
 621
 622
 623
 624
 625
 626
 627
 628
 629
 630
 631
 632
 633
 634
 635
 636
 637
 638
 639
 640
 641
 642
 643
 644
 645
 646
 647
 648
 649
 650
 651
 652
 653
 654
 655
 656
 657
 658
 659
 660
 661
 662
 663
 664
 665
 666
 667
 668
 669
 670
 671
 672
 673
 674
 675
 676
 677
 678
 679
 680
 681
 682
 683
 684
 685
 686
 687
 688
 689
 690
 691
 692
 693
 694
 695
 696
 697
 698
 699
 700
 701
 702
 703
 704
 705