Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Фізико-технічний інститут

ЛАБОРАТОРНА РОБОТА №1

З дисципліни «Криптографія»

«Вивчення криптосистеми RSA та алгоритму електронного підпису; ознайомлення з методами генерації параметрів для асиметричних криптосистем»

Виконали:

студенти 3 курсу ФТІ

групи ФБ-73
Дем’яненко Д.

Проноза А.

Перевірив:

Чорний О.

Мета:

Ознайомлення з тестами перевірки чисел на простоту і методами генерації ключів для асиметричної криптосистеми типу RSA; практичне ознайомлення з системою захисту інформації на основі криптосхеми RSA, організація з використанням цієї системи засекреченого зв'язку й електронного підпису, вивчення протоколу розсилання ключів.

Порядок виконання роботи

1. Написали функцію пошуку випадкового простого числа з заданого інтервалу або заданої довжини, використовуючи датчик випадкових чисел та тести перевірки на простоту. В якості датчика випадкових чисел використовували вбудований генератор псевдовипадкових. В якості тесту перевірки на простоту використовували тест Міллера-Рабіна із попередніми пробними діленнями.

2. За допомогою цієї функції згенерували дві пари простих чисел p,q і p_1, q_1 довжини щонайменше 256 біт.

3. Написали функцію генерації ключових пар для RSA. За допомогою цієї функції побудували схеми RSA для абонентів A і B –створили та зберегли для подальшого використання відкриті ключі (e,n),(e_1, n_1) та секретні d і d_1.

4. Написали програму шифрування, розшифрування і створення повідомлення з цифровим підписом для абонентів A і B. Кожна з операцій (шифрування, розшифрування, створення цифрового підпису, перевірка цифрового підпису) реалізована окремою процедурою, на вхід до якої подаються лише ті ключові дані, які необхідні для її виконання. За допомогою датчика випадкових чисел вибрали відкрите повідомлення M і знайшли криптограму для абонентів A і B, перевірили правильність розшифрування. Склали для A і B повідомлення з цифровим підписом і перевірили його.

5. За допомогою раніше написаних на попередніх етапах програм організували роботу протоколу конфіденційного розсилання ключів з підтвердженням справжності по відкритому каналу за допомогою алгоритму RSA. Протоколи роботи кожного учасника (відправника та приймаючого) реалізовані у вигляді окремих процедур, на вхід до яких подаваться лише ті ключові дані, які необхідні для виконання. Перевірили роботу програм для випадково обраного ключа 0<k<n

Значення числа p

P=10135538012407626858726694771979917848642343601241306598871364357157668729651 1

Кандидати, що не пройшли тест перевірки простоти для p

90185745186582484012431069302109717433682393124464198202083061348896417513471

92884909479496225649857867470764381278496867326516667961136953907377780817919

11283659937526653418589255782899179538106840592633090909107845446416314925055 9

94868186569517773636644070014505744409878314567214111017979529488965228822527

98437900719763542069560137401950672556905670723819965475365553580797518151679

Значення числа q

q = 93355627360367236277216721331318062881555725369696294768340295769881619464191

Кандидати, що не пройшли тест перевірки простоти для q

10691107190558017603700427199986727813176266709830982781775372332038802846515 1

10859486291036166363026840321599089792304710655966351891997477725411799256268 7

11139303888901429227957672520650794983210559174662440313853332646532619855462 3

110979173914216287519821527334927459659130946886902589167284615354858197745663

92905260605448487199585566399383506901989749688554534401158869849569157447679

114969650079081539632406087071974064444491730907738640355839127751184963600383

10812129879811335712624174916676478545150028219728876528160681347684760761139 1

10683755624416117987762463295147594218014167285201275010159857877254679193190 3

11158296265010177269010407120196885576134264972343705769258439859893556792524 7

10291240374899345067914937761802318385225026719512869614453830857370574310604 7

11440678981164766015931566216651832271878114232319734768442850544667557101567 9

11092556222715664915726531249941342925685455709352892751716309129319810072575 9

92466735060894699120569627545745785334046984991954231694555510283999243141119

95703115270103582607241903459268426399963967678148436480857822282738986647551

90996407034825390916689267054517749801571223669820660142844155794377511272447

91030435897075413049725056763013513684007717905714239807300464881695677480959

91121482467715711351009333185945013759402081533098705696711110927367593263103

88314263498894788968295029626388109418617063938139834666549541428558963933183

89447301461093462477187526401066286634222618437518783992420064058176998211583

11281474524512596623802078165420219217168715104962704345811328536882975657164 7

11347062257022458445254706002527895613482446530218087408871514853533935258828 7

88829979848056644825117901321277355588635028148725132609513203119439829008383

10288368183186895470480742659964916091492941503061539492958256489801284635852 7

11032124257325512216257615433960344446850277870914060605779482626142331364966 3

11399748379130621443239416863611702003487435563790396820395763508314462643814 3

10812183555804908205562675271513622813531305723790343099685138430770528701644 7

88745100199151944475265230124934456314618365115757316651957475780330562519039

88605027908919201421553231847691457662972438640588498697436330578786809544703

10729467177047696056686210391397339151754785806057121523153247247728154299596 7

97522892425120679955451869729201636408459566306796693431688876839192714280959

10625204390189523449675052068225417820124521981323781892787012594573055623167 9

88180815911884386275191114593567725902523642532890442595779338886894049034239

10571126960778973141922780726585588943341591093844068755928353637108313541836 7

94256027502959196667176021206382159261822816942422204548045089024573653909503

88051383750798810883336999093512519618937189060050729645980179282292300054527

10534116266115250472868710499997310620858897975968969548496575141959032425676 7

10467362202272550494309153847967114519064624725479943358150692764077179849932 7

93009991352103540282791261880409414759012945856108716009855641646703365324799

11029584336731764850394626242301804172666734082816253137938849474473436905471 9

88046708259551840939656134950601222245526611523260088670076359722219641765887

10141732739005225321052059447543263839598377609672074860808735497215126011903 9

91573762691572073681440557788170689379675929099291326930937137228079498788863

11239543960180424915677010733873064955171265164448783476280910864006607431270 3

100097161245677089743808929065929592440330718123576605081479706246443928911871

113163677263563439228008032568605853120214880536386158155559283045296274145279

103140113989083724077190157853060974504786826950537457231363135399433252896767

990132039838591770970412265377708011899727090613774131965898123008759111352 31

102522631335103503110231470260238485477336621656723633261697643273147438333951

104080631709427652611740626056776813424207857537336322824686857152952384094207

988424369215814115590302954614067353094586171582515236682130485700668497592 31

902792282202528604351300049369473477296352997385696969352449564569035373281 27

111712587702666780953853755348010445816661243918890708462230381708383048695807

108396631469384351911143216499379923226172874334995951769237180928271805579263

111078114024350951832433918413365355678200979830798311235612786396130330017791

906770378922713193896410375548765955297094163126141972496388421141001226158 07

890200299560917596367963119616543486280823070432688837024965166291402379755 51

114654764936823798832223592578107740044456174717358746671300395346793798303743

926800380840643007449444827335670598760206293176279514109118738689054212095 9

952186305608341359575491434075841913711717021515690666126625566343329103216 63

106804222969079770282911903749395774069184412493285075838433686458959688368127

111960471441666847845996181433055674992403081481731976615598872071740413968383

101655298384511451490836922129675845794609242967582974871009764270021049057279

915519176620331901737357648999062389750776392794091130133446584532855517347 83

904259735316438984186019735663405889851233539502645918894011607681646932787 19

878396323161191604885064223196032651581039684893082197842370048163425602764 79

107033854850603620575322144942871139639605451888377932716071971298084200120319

975148450900204370215449895819003247180782784272465113742007268467383436574 71

104296693349981227667668509364716362612277433224900488933257508855304444444671

874966759941356452924193878027238837872733460720103803750133031968826531512 31

970060745186229860810429616016325556797688836483549155509834292994217596682 23

108922277136508460530434954850205097078141371020562961422396206074396565045247

885231355680522839113676242713898666337032129451049445158856058236968527462 39

893508941809349728919492476706712240825969598987649006244472974491247989227 51

978259180897484565010587610426802204288651474716778152665271403935570361057 27

114563526389330149367179225480212358909541419294551512160599179064175975464959

104369327755979947102215201656140139512895902899702540565996574793093204672511

111632454644293240650344464354159713619944603444489822485774456627479739105279

897237844929453203157095455883644861897925754618736360689544513007009472184 31

895387300099634168363473654063529781923447905394593458155361282502277977866 23

101653804132613346559128659080676453423300558749762302627144498029606513672191

112035298580065949550515151780750256631989911673908572589131722285806683947007

918713142513290120608588301201649583814734494803017351564952208384035630612 47

996477485830325863222438669206731177309833927434777066388759328590626784542 71

106185282499324178431541598863802590201406963812358428955695257772978306285567

100687697071569111729722754487571449046227032273828439131149073775524852006911

933928556608447028951687122953439699774282362778822250804150967672495931391 99

905442194387651792198026777069396155450085982886659609229373591011423923732 47

110795468191041035970469959026146372334769737751394826628586314028056053809151

113621060197173609839795906790669735264917734470403298644203258963825140957183

925264436011977574968912616599235171044198163358636731303568504250341177425 91

878125614070538269966128370168450533638221772572066101778317202316048612720 63

884187870692062089822210586172951070928145431740930847295699158929942931046 39

110719841013686177247473102392077569716703395824907678671218135665314807939071

104027653159626612494322589466238740492529654250183618223518543501307966128127

962316891486242255170984421955474641977033543287250197439901800081247751372 79

989632891070750723907859392163388993343119090316181899978270619445547530977 27

106613003149206631212378546195670057657345164075636331984237238540654071513087

105930581823349947390764932684062499960022436001423431159883669287471203483647

103926695488259169836744593127626944348111314893881293909559474855110707773439

976460637553892255954662674116778089430640368559851008616138340256251631370 23

114708942763400139601494806254868654367290692304291599683712224952769243512831

962042885758130968993351759402864743038143681588755657004135977771620892671 99

106639509339198485608769501672962218779940354944113723098253322195201905655807

111306708933279064269378720631068113369893811088448951925109886577668610064383

980454800609331089582939439771588069912241852667059203402115456065806721351 67

103229430171486264258228875546708563902670964313460080977901616603826478907391

111600716039337817115145851418808584862154972808811522342092845950679352606719

104047223929293708523532410359831172216202420153603494951835135226648473894911

107911558587493748204306223855614331390914125368234124759845377899764413104127

107516334929437782865496086754246766619630729203206162673979556231161193693183

873564709951074445092017503095480839374576388867280383685461126364013076152 31

101882281779791458792592292482292947703019073720579351393987971216936280260607

946519288664593577843242490598297117684528980826415955331200638680466823577 59

973418399174559445319738347747453421656218896101417752870522224152968828026 87

981621226502940789942786379321001516558562870749285940492112411646485625569 27

108892091639615944062059428712500295708567257433099218980429496223750319767551

110047272600515459702967387035264110823208854639521266166064074274590007754751

940106571340456605190899703650822938816487003065430775845395219770334961991 67

967417599432282537349083850531011260540925236311060794318019472409536956989 43

110942063345623639897817458688931689369673871228621723330970397412032352616447

111656370787843451126762497620308396560747408042295644539825647215867307491327

973259973673255959897915179946340267846936802890188471100889376441654250045 43

115555285503500685713003384960849714326731041180612333919424084076821728985087

10375324733176321776061503856956473614100594329596011621488624837545848890982 3

92000539454293931407321933971915273267513499296056124040194191278308262412287

10936257445347010770924529478158777275883210250573710307028804368442537594060 7

10959853029043717298995994711474777579668563750303139810918904146515823322726 3

88539649699778536299356620215667891078498702049541183359045403723050724098047

10989146301144662254085031796853965846170079220030676193853652246093982649548 7

11564094584031296095487934055225485407391739482556618214344148958974578786303 9

10011925275163949364790626304268163213064738221564225257609107590058708513587 1

10573063835734436002534737429191602385595476299983258679457727762609735729151 9

87905650194665761596483479024428571152610677579820860744253953621432077385727

93305260842289966371353361929734499010017760335737095953742620258555333181439

10578742824120387475146953938774710984474828583574183877483898692959277166591

10707483488605999671468418751773204046649391733397002365033504838624650408755 1

11153662120424364878304264057609201416112156287474840135113792312645037064191 9

10355014429747355607603138020969304132224862454964209244446285540050806715187 1

10617857674474113928162008525556918531518509915733266135866059352734725203558 3

10490336841886765212815234657661419224146548343288264189442163624668345401343 9

91041656165299578259189129251027626349130009540679258125608724070824597782527

10936691515737592298639270766035735637647316375256235308443581865940071376486 3

11384077042571314333815123368858605052979727430948781399445887659885322253107 1

94842044325898877275110768965503277337493981848370298068201021187759624159231

10763400106874344614804587019503661083672238471449503089989954865208637993779 1

10152991487874481009009520146550680977562003416531084897056294014555837693951 9

97469907231779221886562642655085823771441428761901896863773074550520994070527

10224740065957787923408435562312719790438946504932489989773759033724093281075 1

94005520084119858726452460380594480976825205416138540155786950480847886614527

92104289951884589357973745413363862553246902212886820214530319801322280321023

11226305055103540685061394056659932671857509755407297494813921549419188060159 9

10653530530192713285881017618910300310878038750923346253252093938532725712486 3

10795952772679889212344568781983350994294447108967499037408451286371192432230 3

97568407566581118477428818814944503767943116949240717327784323146550389768191

10013676535669508287033103230943941531793388916621725104328819095834458062847 9

10382211403511102893930735821729817603740118698479667058796100262713020645375 9

10880311622033191720146718301683752747436103669233549471763881145019430810419 1

96428383376313105798949406130015756351702678773895543891807789052247440097279

90071531551463656732188375491865764684021269022218047668754317865626653163519

90881643004050177393680668674371869336810294179251178501167510303932038512639

98058845716636597323108125074199235063470475683120176808024718251914073997311

10642997338512242110117486964983229061677965980912152198744118154315982648115 1

10580017503708376267455929920609740554432993364629107294985994002579348416102 3

1058022916596719506880998335509486351610103027343877985429819387490267561983998787663666817168944602158241984324892864968187099254316198089183535269570150310266951131609814329459638932265780844661297570094961399198926652966721237811191672464042363066468888027741422464605623369867814625492777984019409930289151102378029559480608816622613689574940252603124313716622801987274034125199114239109369157261525574944671995349558811845940557861391731752041695911214105755647110718539296841042940190931331873504471236028708841185161609661884204161433599105750605363873604253315682066360366935284820346019830801422750032839504297983113406414981188227316320536356230885758550719835476367655163524758322020351999951917277794889580236487426687730645376218132588853189104359797137657708868631138365697214521528828707744220316084174335234657001898842248647406850460876799372342481667845822869529392890252130138748250359517031796547940205682963251194164751663559550184862174075264532773973347752971520201461747049374777606143929616411124605332820078703680623002301176623615154575743794790166830587576319957537062861579334467234044082403946104243741055781287364015469522747968716787504465802200440714586803057992602247607604854583708825710538531673630310388608254968343741710133676551449345386388552256819820593120261924968151908351105208244873703189731019594242266074129373505394237348171444237705359424225271110266919264497388081695686565654836498267720706312751162015089568518242303991026522928920200285821527801569507743390506293192126375219883279910634331832318889604622282848157586418443620611627235890957760127000709304835086230539468796100855982593813070136758823925212633832855188405473800400017570866879528959907254534359347432703980269268591770001868233490070846075562320221183693291511112467446893098135590561085955242140091560378128744321242480986146191756492799224951860504091962961719145009813178365637684492990462193381582952089832652711302171894380895057609413550259884464392422010191334717795846164899557487411110132065554127043125981943616355963891581928328203692568585800586309602272870310369804705525708879459825022115347107269159808795791727568918573378022303334310596308712223597619513003196515028660786154396642566618706522918998346878156797125362240970198369131418835014435557125280497966133694737469235361501675519999532523521846603422436473140127951967743779520994743822765300704820320010239729957280449278893369988613948141639717496224462325870458171061678323322060710853188197046860469733547132990215674884688362015045120666229561930225496883110918149372941428215319376603420002308477492348348091387958213222846709314355110161762176024443459219605517477347864845777462881059358005713891681023243059111157048460543577136071114325774834036818220708888886670934721023033046414131111314252410505380374973897873569647362444234007035570201050587294278020707123189288479246739738728937272208801798551476648943253624601691350306260294565887945587329425069722978457503516782171610799140812238809129583051932322779627519995671769611882659092610105459565106160337870022237821731403233745757536255910

Значення числа p_1

p_1=882652470739783778379468784849261235224139046272169612450344389519705248890 87

Кандидати, що не пройшли тест перевірки простоти для p_1

1069759323578429330257996754302194163163668922162778892606326289701785976176 63

1121864839020004128031230774292944272868148707818554145079994842808216820121 59

1088315166048146344603377344080904206010385185617393956661206283413157314559 99

1127978296551551474535475999562283819991999570170685812318677912620652078039 03

1107392940481118563828781139362434055440850285756724729551127461370503081492 47

9565264768913480455788518561096448643766637548609934620735121042615735366451 1

1144610144207624651537977056421567139595044015216514207837308215389924391649 27

8789443776760998522169185396432070422816980669578139851144548980658654727372 7

9203548746717380800103117521177682270603228717093984922800071484916548134502 3

9156580169097280350703867521907493328055071415963646832597376885603154735923 1

9424799495823897536717503816342428012333580769267867655996116515606505717759 9

8825931490944035856775849989907679174051335670980838453739609807712472268799 9

9325700611341141378212825447934038099981855302051255848118613724917243523891 1

9460813325630457240125344799358526373779553897604246024961500185013435708211 1

9106681742730463899140353659242543897513738906799779155613590458416968918630 3

8731321836276689990244490771944815147367362555592397858935343923816503967743 9

9888269852957946132023067991701266893438817236110804944340009703124945495654 3

9646444270865994294218471583385675043819558484060267450162626587946318718566 3

1003605715140767966519671428455702897972317947363339391279175474776800598425 59

1052420457181832634422616777140567246578385777528325102855388534313504158515 19

1135581530515181182996618217682732832285813199145969847389847335113997764526 07

1052930790128530578229555734784335159461778849548177349171510867097001310289 91

9523877878900891463251295917808934804598735420129359582248783448198098400051 1

1125463733381660319409012641933207759428979940791472290962572118645926873006 07

9501762454179639331026886322816634184221837455586415725455635834416280056627 1

1090240674335652524303565292873954555508157700270205829851509770443781224529 91

8927641806092588079360858916305168885352684658923364893460739221866373395251 1

1052762507948703963415930846981836711684786471658001619109957828879724461424 63

9440999440997233731237952047084107195026966872823345815076539512003964069478 3

9014318711820673905082245951843933568119406399410564078784635227507850739711 9

1145249134687213175075448413110398840337411674105557512688264300140930531327 99

1018038988427743396582074143882937634967663875076058886654877602948176590929 91

1038351817560235621965360788088397131794426710882105657896059587777799515013 11

9202506558435197264624145414726655572683037085613384637181681691961693359308 7

9538736414990012604399115172220824933555545508592712506397741600391557441126 3

1142815225495978184751315504482553656368670122139220527993348727174012390604 79

110885394725147611410731716031696934826891642672356240889428325737314002141183

933683008503792143289902895509710945339783055907879176701280676721542331105 27

114000406773704331225246087205290208453988693703360035774896282198713181405183

905277591921098494364791814660864249199577692153583273975296513563147894783 99

908035619593762385101631419441197288594797637103659957707429338314697730949 11

902462389857512026957023042523735572033731824637219030020120153367713960427 51

105590285150393565647639268785585918292253821422810108299250465891071410831359

920032426745663620745815368786053912270361654579955512253642401455364783472 63

103189718932729155387312649167263283509038832726178775239747234324200831320063

100466194370032476118236416253184201340831340211598954378383351169665216806911

918623840610676601055254866436812546287399578530836668006603056658105898106 87

114913265321782920400068476452154608575162263747619320565797711565711269167103

936590198981252206422132142416104241021830510335644436979821370916271470673 91

938621886152064535032352261223970991092617413287270311245936772248085861498 87

927834669370267089715685420710737323272908772429421987997705276829374684856 31

114135839825101735492122340531873995607022112450478483683980064255759324545023

113798146868937456657227529886289155627613220619216913700161176474121492496383

104455776033459108396382780042580961584550729100681484245312946266135458742271

100244943318647150796319707432260534449584655972005647359324020467827778519039

989502819245339007225966758325925378705173223281116358711483826422004989296 63

980208257661622672510203070847639808110749454944820184204292375702216208547 83

111990619071894013463516781103686085370756208885164383971742473986327319674879

100752810982587879054453278690487830726940966529462561592287449429577322987519

109479701142842806371633852785604580846880341407131049517926306846386335776767

101754017095790518980814109574491759794875801228396219834668563204478934712319

908542308771993960382410894590859209931126881310659359487265768292699414200 31

113765137544841629872340713990747130459448299443914801940884404153608111652863

955686755413255508218553510887791866466243318977969103437524068504797781688 31

111383942411859735770823014134773123829893828589639059617473617219380809039871

113471461001241509946121219902245078060334692286380682553488978624757643608063

925525511284601235152486959339646370833799189552080123542142521517378686156 79

104801923954061649701814956894852662052442162541321802261335853787672970526719

944151907718045702362230806709567908337021867329218235066494849846162353029 11

111745618017305103224272448005095996101105552291901302290362166114456795348991

987931327243016107787203832257701832776614874018976131849259591297819675197 43

110179138870911800954757660269781938343829690140790563543929583774078024548351

949235863272009159848246450048398909474774168374141955766476317923476798177 27

957450832993218755868034715874155566300012463627749151167209539941927936327 67

950216160885716821784205173511668959892843310886345551417000942546037914992 63

112284539922493540683181276188523924277419935525973247353789909163168665436159

933262260537322306777258774041033836431431106280262928906701045902693577523 19

102824530607286617492993523445998014119451564475020835625979797687642903019519

949715170706974797708949848760978021138956092966319768843869162677723610480 63

111096999398399538957050058975513495424504237570131491623359745432893826007039

114840539809579590318436684586264593055519678725148961725955957519993163218943

922243029530910223273751814005325713745102356788666247379763672915463991459 83

113820690406753937389508343314224229388241462046110730113892085112856624758783

104787408231127154982982249299243070765936332488299642546646661119918068465663

921968123922720171691582165768450651193383318025913796268270729382530855731 19

980110769970488783416388553202799793620387998734459992778608717002701673594 87

109102008783312152859349617488007344680114055118678672348547525867627295539199

895914955967332147264852780477831681872310465781913552779723031993280102399 99

102082227337722349043387272302135980405723762429126217617018398447688428290047

114840239510467510635860972044667485230479674995814457259749015339931694792703

940046653946987091676559018779448496135615726016937904114495281132498206064 63

104482408161249648281957542004103664966609888549700138484001355804664418795519

102738137266777020815159803338915699128877944624463065519439320100391304036351

111932723662536736411093715033999944163312371286214859565184248413438076780543

101965479219287070561909620787087806915835628679835646955530608415949843333119

109122510989863985609229227743510639008799779482153709415322422580060610363391

982941428080934235604602715719630460935210866603181042493623575116568057610 23

968505469968850206024114228226272061527765238446239296721144110983963213823 99

100543483318683328556568849905828478817567204188682183910552676771614236344319

107012331314623847793304820356255548437554279110491705523751383070758559285247

111481925707228133513831617213765230501208948355853604704852555275340972294143

114379948533848681383610133820449341686841960362908760239456458695874281734143

869249503679646164954638775685671185612749319945727984177382228659692403425 27

100642287873907439936593138214174573052513096051158102174774117029365533376511

956361739630660884023921027520580022456490417500439781780588389897424024371 19

109731175799960361875341210966229995199152989824726836811436290188412832448511

107795946782140328395793041440930490235282445642088100654043427861322042179583

955626603402707744204933405685245137175782488570742561208105021580401222615 03

103862197704895496074590787320324158166771720253166749094803844944185060753407

114525293756690119246060859375175412455896536119236605739977091648818495094783

114017601519024085144220989908984717161429991803459467286168952786809058754559

892134491176972404091008152852579334695613969414440670654579943698830217707 51

899219595979560649569776973710635553124927074580480182872881028312485247057 91

108424833098394795107111238125755100569112531894763718241616637638833885675519

105073541862461667677645728506262128875454655262338181874679591666260765573119

103944338684134661290754131818899363956233657687914879352030441376559333900287

111198126702173830344024092650696485755576681546996739107982535501208115740671

914425379427915000393078219671351979862560788597625786851634540722185190768631

103292343392703057167168963755919138844304909818897863846150062965015390978047

882790127944157930141658740281365906299916178986598645759946422752932192583671

105018256636123459608064081305395253740978019398733289980008929145831504740351

98733009314045945223835210277891436923933578161950545707623799210498884370431

97827936581080156600455570587287352686221203725888504275566221000732487712767

108978716401295764023852219300297489943946736978542015949441386287959493312511

105600488318423572980443758339542451532425657100960087601934615614502775816191

93709336830130680989932454796777994862653571446200012878444807803315316326399

97183526240125579609323256001716450772330286757334406376129492577955363160063

96048068021116034733527933096801133546165330358489687348325971871651409690623

87342410741762369500890543158673242717119208033948578819523707413269716140031

90498410708750127998190075314483107238039804374248726208931003281941400125439

92850603778585953253277126382502936324908733277729776842761032326239888605183

97592820274871607412814846843266341300343867494703479917579564599972997365759

96666185963737513671554741189420053012159059988493419964641533131787578900479

10790637715432450129361843223424738836507459024201731323945183397030761227878 3

10490525669988459328621942544067361310793497613320769711321702083843249615667 1

10075466264185897125668671417296888989459769973682635089221816258596580622335 9

11386276497594879103628273362591336193746086710068933860433322541106791343718 3

10174471425312769693770697599784767170437563779318224213669264951931368059699 1

10738879379850449156790877071775401832424312866641639963280494778269641264332 7

11069573551928035309262492574324018141348446612425879569196216841005400116428 7

98599925397785902288838827339943112105493193138539156225576807974692137730047

88595256963607021717663769858200838314560341988150125493318502246671471083519

11565686417628928641633818349195885471425979699581182178907857110903755374591 9

90216005735169265407139649763598079953158535753924379511544035829054986780671

10287575755027988733626455161650841580109231834571068963363826969495326828543

11393016712328670399120480361806164083765012870866352975061888972905327571763 1

10586541208142299701326823159964824174888089922703140371689381619454451908607 9

10916211172940658566695782103039453087472797014579630500482940271552554545971 1

10697555896745338562473156232618764057692298119179666042282690951098787482828 7

11221518797384902217084941390740901335939520800690038699298242066093108835123 1

93718750706517983536870585202786656017827578544091131994577060466531508420607

89996703216324404558416886255157406729414705945450684552910203693465623068671

11036440233832577977816200101317020412858163268284856551617077196887815513702 3

10994099112757434858781405267458637710036990203040706504912732185573577457663 9

88647118907330663892043412378215140122812081182494975834595786872983860019199

10558087491723106453367168048040423660583945098015416943264336654319077241651 1

1114194801540810934845163553274764013637602906823815696510687438733882341457 91

9107418283543575217167866199214966120607461030413277642795195906026662946406 3

9715397117684403579601678892251183188864530715832051812572387085406840448614 3

1060910224143478341400590043992893349341604512710441058135498280694857492070 39

9350211354583586543227455852916207012632425532702556453385241181901908082687 9

9219154584175763945012384816380871728322270259593548452658016000582338215935 9

1148790426551499312819167991985755112218944350667186942489058277886918446284 79

9992664150182028250674815132058870339501897993128792148372867664036577568358 3

9141097663581549255581679549029368844417950957305788789077645729457549344767

1075789022156164635257346008581473662939420943428943855985168549468344240046 07

Значення числа q_1

q_1 =1128741193246458919455407993276423220365078483867326220645809357690234648657 91

Кандидати, що не пройшли тест перевірки простоти для q_1

9901308342262635012840928885479719599011559480880159909114906029916679254835 1

1099097976592120728634180179754761236319692547464713256381270439743164231188 47

9564540472245592928452317136966218434594605125646881610308527971365158282854 3

1012053317005868099023912152401456614127500041153676580139148707320381560586 23

8701868039680992494688156620833233023225348662990838665146946025557736816639 9

8840856910981589638649527160322572708163336476139462682077823536671258522419 1

9956289009979919463586142303796355926159239225518824002298977681407949668351 9

1133327774133333046906649806860829842921310622931176025154961761720119622369 27

1083441536957836156260707249027603769053236220897855695813927655761807272837 11

8986327521438032712262159957430306084803657961111865080924671674708931274342 3

9031075428723141199507382154350166184858510798557861987642062635771192200396 7

1061748760893496013104208255583783753440763892482301105984771913681753813811 19

8756899749056121817521352863435135366787385812249795982997177562649103591014 3

9228608548710111141801632360818666118868759514383949588506648520735951159295 9

1143915318024394560747633517728305164542702460122138377272915868761943598694 39

1112635671773028979851561322540870819506514021860910158062002495940911851110 39

9366736391603452692820084212328530872009541873136639711799629466565830560972 7

1012747180298532499368695425693467766816097669281647614631131190680780537855 99

9672649127246547561055805323768900439811312838963807887887997354861312501350 3

1028475773780631362639387406995352316554965226275488579430944279306562269347 83

9619991652354377462310045656089889722550500117422466658006743097542436716543 9

9455247408789189785067241292285203965181111281781101273889116834409438248959 9

1032123971164638907166702125091735277446634563694350480013676392733540918231 03

8840977003801431286463001717999406678738050557781303733368684487955052285132 7

8723291443188966787217007645099240424740014923212440474566059738652497202380 7

113526801363779749639041364546315292238369367776655466807189337448639625166847

105598766503338349098674052071100868877991867983259924971488044666474140270591

870618654741738627003571584176847718054920598026801627294917974828117288550 39

108815898387810236368459759708238468626850019363486341756772688237890914746367

108430687087065387099007227874593939683294971955341759985520438722088635727871

113052154556679760924319230262957847055913687630748934828276471318323841204223

112261882096135746189737021210063060065715783280188105448651811875753811771391

975640988991040982974037452986744535149461165316162848859658100124486415155 19

112229817185527462692164729947382565104095753860312896395304836889029778604031

928676612312137545261687413986093186610346172031865037573041373520029852631 03

114145746799092830949223317244844753138717612377727237766733479279199172689919

111665745760241785288889040321891122158623767911495392800900546195513371787263

108627966382636448666019954597060205481599703612406376497675879042113247117311

898050084426342895059588687543731728707364798296274671361573226712202739711 99

869793080154575892758746761915062407540121696358261856861104890659969056112 63

964332214159515924139464277717259481030124516948792631447793809465391710207 99

882324665371616864730150781487895602580387227394765965697504806884569377669 11

926809111912787100678484900857754347346217675198400115148525721232519284654 07

971492416211494921937682663060683429016138407380705223661801911695260306636 79

107694007062965055116738930147753560770441777439678360289057626977302137012223

111761030677198614765743097462845988705678280246605023808694617562963584745471

108575478996714435779583714452102892690081014087655825017838370528740239212543

956653389759015156407395399068258244725516421899731131488750419141694932910 07

871996674802015224883569789499169633104770805447233192041639613350058330685 43

869315401147725167541641741232944346000563022697144707703347146230388049838 07

112385303071257147462618159832027860129001486345075638187461797218113439137791

104790056376832827106673912106198523913358233238754961653769265829403483439103

104792255875986964555136991646980671371779475868148451337744907197765635801087

881324375470114601610030632463519223629306330868534230899413981141498320650 23

109243466168456686411146087071243779572300187699417201580403828219705160630271

106703881125625576324385667435688288396465082006655489003527822283347966558207

111278132087084728422517932621018688051170922215025737785614960644880953507839

114832751137057761179233019733592441265925008825395610823551395283742368989183

888414394209822877784780606411209558914982831505898497665196954934141916282 87

101916853092800489864668381681435656790090529057939917112893511201115550515199

113445122177227140016409717057024045027519081545799317648332870944062630264831

987056015023264115020113167632744149159549206307460058479410850695428880465 91

112665436686765539096623249350823543520464278199981853936620562705099010342911

102518659676207602919614247290295664950633937093960328763579688684392376434687

940619361465891101217443319566617355624816074501900349597630201353609622323 19

1054471849466272989741730815528363712851304106339039518851664121057677232046078940939940988310660022951256315570548297409189092952066271097812503992939315111031096830360121007916370481295474038229602653064119001334299720931716418764711540038802793786438080990365373175275850132495898244806777266384266406867763110524514337931978976270007407680225761231881305741004697051786913250546181734310272187679005928104199416062433459032883909412824828007033847320292357255987192515829982411691937598952181071258548777172053344226267491789140006413533183937667976169231216603248049443901092348974099251121983740108334617209371361271027290691773364634141340588638030503874893028089832243027283848880895998033911154653322998716312852998165172130411821195348625894156238129368226260026654719817809336166056612404332968043115842804980440043763492678933498228514383462310571586330801429285945300263314694996745819945846238281940215952964741129830310917629612114701236221567587814106243908806292708575969750566304031686178897919506761818131638647192431518565174602211386310974950168189276667065218052915110503675956883588548525612428014517036380726421783496754521983801832343863295911482542840864825161179005979587454319489825418146714057617459529533840424959993330277248441761413668478835282372914775853609311950500721313990529955921919109201271015543044609397583719389140250913347719364671426775614201356504530943107415761646699574322992819416249274926333792287036671501964130868794314719231114662573915250916288741805285255544777945878903406026823065931610042178469887102453533224521317448807321498372724189545782710855353049133990850102269313023107769402487358938354754738119806161621559025323672212843569353528695712645119110539274727324109099217533726924262520747132852399498848169099868981577121791877047192238447478204684345292605356166140852258965512876685287729266310512631138562517964162653234331257475413089792849822226999240688017597411803770388471146015271828162740408571765956355452229386674825880191754673228306332621209598878803775461430505403848867540456655923482652193465384771601848211138727116799673851625379316781917762195925077058918573093494329590815094898024679211007875202268857218173289073385678801139556306531496826185093574551169050943160311074427773035850355879644232450395414968025828357866202843671148885329313791999691825355818721537062384445071673964320834090731559102717236113808230973439911471476512904083801466576785111366029521879834820445868175970860400930245836711471476512904083801466576785111366029521879834820445868175970860400930245836711433721055340194038124858621534518139535229196885644264543349804911594727014311321644230241305248613247005419019947616142195945468672315505185251005562878757336963644730563481013019166288027925337887747259776488478270504803762175976881655808467214733366071768967915478371055527354269646396098673809377198079415270615104929910215484534596745269783183845813885577305851450886659558604794152706151049299102154845345967452697831838458138855773058514508866595586047115685797338166999808875553336391719076684166999082844641625499694658805039103

Параметри криптосистеми RSA для абонента А

n	946209509783163606692481825203095190263563337514355014991464547207318433836585767719918664927484478691882338129843370428481051739939153237337578416373760 1
e	65537
d	299930670556104807443575650957002906488476200825842059926952477282830052405829199474596673058982820116439216036012456129688728120900548781586934228531727 3
P	10135538012407626858726694771979917848642343601241306598871364357157668729651 1
q	93355627360367236277216721331318062881555725369696294768340295769881619464191

Параметри криптосистеми RSA для абонента В

n	996286203044758708763262433690236694514575743106276754318332511064028907380089232466775648784354223138842321318485950388048842133252436711441325161552281 7
e	65537
d	220199362972112392944990712910311404550767194697566546932283235771586412612727179790179850548340472408941811784257394955699021504321796357705482935912917 3
p	88265247073978377837946878484926123522413904627216961245034438951970524889087
q	11287411932464589194554079932764232203650784838673262206458093576902346486579 1

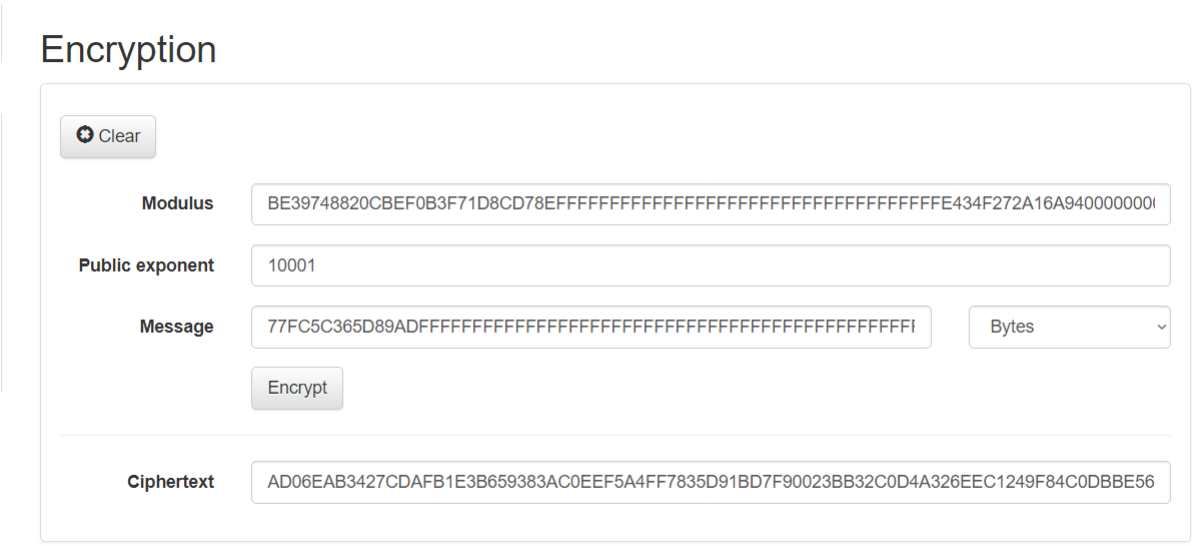
Чисельні значення прикладів ВТ, ШТ, цифрового підпису для А і В

А	ВТ	3292813248397098733005204441233991168918445997272464707778395459250785615872 0
	ШТ	6969791709667043378391415573171013419150400774760490172249971218102254550088086498346221995078434740027994908403398553525003050210801709726090365386391295
	ЦП	9325181590664658794189696255036890489817257086677498759790508247857011262652032189829308185302529451684240862485094733014826051958880763370445284004310744
В	ВТ	4590018473564170820506915316280971402034743944141019179508955892558417048371 1
	ШТ	7189564306336296841605501100411338500006751628862199387390177163214935949754902613622855615883559481185059691531528712382706613471731088053748492028063115
	ЦП	5954724688053983393353920653269570948763055667850050113159272013140363057324630270182688569907180541108556148949845884148229707791278614154138698755157233

Підтвердження справжності

	10біт	16біт
n	946209509783163606692481825203095190263563337514355014991464547207318418433836585767719918664927484478691882338129843370428481051739939153237337578416373760 1	BE39748820CBEF0B3F71D8CD78EFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFE434F272A16A940000000000000000000000000000000000000000000000000001

e	65537	10001
m	542711108680845590889673296765357578880466840776084918493871806736516052418 55	77FC5C365D89ADFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
c	906216028896340355795201850042987465258445507097162723258031506395758811371380207776935011886843103087174801357750956249825225644841780493562789618384193 3	AD06EAB3427CDAFB1E3B659383AC0EEF5A4FF7835D91BD7F90023BB32C0D4A326EEC1249F84C0DBBE5650B4987CE423CCD62DFF5A66EC561F9033A9362ABE48D



Висновок:

Ознайомилися з тестами перевірки чисел на простоту і методами генерації ключів для асиметричної криптосистеми типу RSA; практично ознайомилися з системою захисту інформації на основі криптосхеми RSA, організували з використанням цієї системи засекречений зв'язок й електронний підпис, вивчили протоколи розсилання ключів.

Код

```
import java.math.BigDecimal;

import java.math.BigInteger;

import java.util.Random;

public class Main {

   public static void main(String[] args) {

      BigInteger p =
generatePseudoPrimeNumber(255, 256);

      BigInteger p1 =
generatePseudoPrimeNumber(255, 256);

      BigInteger q =
generatePseudoPrimeNumber(255, 256);

      BigInteger q1 =
generatePseudoPrimeNumber(255, 256);

      if
(p.multiply(q).compareTo(p1.multiply(q1)) > 0) {

         BigInteger tempP = p;

         BigInteger tempQ = q;

         p = p1;

         q = q1;

         p1 = tempP;

         q1 = tempQ; }

      System.out.println("\np = " + p);

      System.out.println("q = " + q);

      System.out.println("p1 = " + p1);

      System.out.println("q1 = " + q1);
```

```
        KeyPair keyPairA = generateKeyPair(p, q);
        KeyPair keyPairB = generateKeyPair(p1,
q1);
doingGenrateMessageEncryptedSignaturedDecr
ypted(keyPairA, "A");
doingGenrateMessageEncryptedSignaturedDecr
ypted(keyPairB, "B");
        System.out.println("\n\nSending and
Received protocols:\n");
        //Emulate sending and retrieving message
BigInteger k =
generateRandomBigIntegerFromRange(new
BigInteger(String.valueOf(2)).pow(254),
            new
BigInteger(String.valueOf(2)).pow(255));
        PairKeySignature pairKeySignatureSend =
sendKey(keyPairA.getOpenKey().getE(),
keyPairA.getOpenKey().getN(),
keyPairA.getPrivateKey().getD(),
                keyPairB.getOpenKey().getE(),
keyPairB.getOpenKey().getN(), k);
        System.out.println();
        try { PairKeySignature
pairKeySignatureAfterDecryption =
receiveKey(pairKeySignatureSend,
keyPairB.getPrivateKey().getD(),
                    keyPairB.getOpenKey().getN(),
keyPairA.getOpenKey().getE(),
keyPairA.getOpenKey().getN());
        } catch (Exception e) {
            System.out.println(e.getMessage());}
        System.out.println("end"); }
    static class PairKeySignature {
        private BigInteger k;
        private BigInteger s;
        public PairKeySignature(BigInteger k,
BigInteger s) {
            this.k = k;
            this.s = s; }
        public BigInteger getK() {
            return k; }
public BigInteger getS() {
            return s; }
        @Override
        public String toString() {
            return "PairKeySignature{" +
                    "k=" + k +
                    ", s=" + s +
                    '}'; }}
 public static PairKeySignature
sendKey(BigInteger e, BigInteger n, BigInteger
d, BigInteger e1, BigInteger n1, BigInteger k) {
        System.out.println("Text for sending: " + k);
        BigInteger k1 = encrypt(k, e1, n1);
        System.out.println("Encrypted text for
sending: " + k1);
        BigInteger s = sign(k, d, n);
        System.out.println("Signature of sender: " +
s);
        BigInteger s1 = encrypt(s, e1, n1);
        System.out.println("Encrypted signature of
sender: " + s);
        return new PairKeySignature(k1, s1); }
    public static PairKeySignature
receiveKey(PairKeySignature pair, BigInteger d1,
BigInteger n1, BigInteger e, BigInteger n) throws
Exception { System.out.println("Recieved k1 and
s1:\n" + pair.toString());
        BigInteger k = decrypt(pair.getK(), d1, n1);
        BigInteger s = decrypt(pair.getS(), d1, n1);
        boolean verify = verify(k, s, e, n);
        if (!verify) {
            throw new Exception("Сообщение было
повреждено и его содержание восстановить
не удасться");}
```

```
        System.out.println("Verification
successfully!");

        System.out.println("Decrypted signature: "
+ s);

System.out.println("Decrypted message: " + k);

        return new PairKeySignature(k, s);}

public static void
doingGenrateMessageEncryptedSignaturedDecr
ypted(KeyPair keyPair, String user) {

        System.out.println("\n" + keyPair);

        BigInteger m =
generateRandomBigIntegerFromRange(new
BigInteger(String.valueOf(2)).pow(254),

new BigInteger(String.valueOf(2)).pow(255));

        System.out.println("Message from user " +
user + " : " + m);

        BigInteger encrypted = encrypt(m,
keyPair.openKey.getE(),
keyPair.getOpenKey().getN());

System.out.println("Encrypted text from user "
+ user + " : c = " + encrypted);

BigInteger signature = sign(m,
keyPair.getPrivateKey().getD(),
keyPair.getOpenKey().getN());

        System.out.println("Digital sinature from
user " + user + " : s = " + signature);

System.out.println("Verification from user " +
user + " : " + verify(m, signature,
keyPair.getOpenKey().getE(),
keyPair.getOpenKey().getN())); BigInteger
decrypt = decrypt(encrypted,
keyPair.getPrivateKey().getD(),
keyPair.getOpenKey().getN());

        System.out.println("Decrypt ecnrypted
message from user " + user + " : " + decrypt);

        System.out.println("Звірено з
результатами работі ресурса
http://asymcryptwebservice.appspot.com/?sect
ion=rsa, всі результати збігаються.");

    }

    public static BigInteger encrypt(BigInteger m,
BigInteger e, BigInteger n) {

        return m.modPow(e, n);

    }

    public static BigInteger decrypt(BigInteger c,
BigInteger d, BigInteger n) {

        return c.modPow(d, n);

    }

    public static BigInteger sign(BigInteger m,
BigInteger d, BigInteger n) {

        return m.modPow(d, n);

    }

    public static boolean verify(BigInteger m,
BigInteger s, BigInteger e, BigInteger n) {

        return m.equals(s.modPow(e, n));

    }

    public static BigInteger
reverseElement(BigInteger a, BigInteger n) {

        BigInteger x = new BigInteger("0"), y = new
BigInteger("1"),

                lastx = new BigInteger("1"), lasty = new
BigInteger("0"), temp;

        while (!n.equals(new BigInteger("0"))) {

            BigInteger q = a.divide(n);

            BigInteger r = a.mod(n);

            a = n;

            n = r;

            temp = x;

            x = lastx.subtract(q.multiply(x));

            lastx = temp;

            temp = y;

            y = lasty.subtract(q.multiply(y));

            lasty = temp;

        }
```

```
//       System.out.println("Roots  x : "+ lastx +" y :"+ lasty);
        return lastx; }
    public static KeyPair generateKeyPair(BigInteger p, BigInteger q) {
        KeyPair keyPair = new KeyPair(p, q);
        BigInteger n = p.multiply(q);
        BigInteger fi = p.subtract(new BigInteger("1")).multiply(q.subtract(new BigInteger("1")));
        BigInteger e = new BigInteger("2").pow(16).add(new BigInteger("1"));
        BigInteger d = reverseElement(e, fi);
        keyPair.openKey.setE(e);
        keyPair.openKey.setN(n);
        keyPair.privateKey.setD(d);
        return keyPair; }
static class KeyPair {
        private OpenKey openKey;
        private PrivateKey privateKey;
KeyPair(BigInteger p, BigInteger q) {
            openKey = new OpenKey();
            privateKey = new PrivateKey(p, q);
        }
        static class PrivateKey {
            private BigInteger d;
            private BigInteger p;
            private BigInteger q;
            public PrivateKey(BigInteger p, BigInteger q) {
                this.p = p;
                this.q = q;
            }
            public BigInteger getD() {
                return d;  }
            public void setD(BigInteger d) {
                this.d = d; }
            public BigInteger getP() {
                return p; }
            public void setP(BigInteger p) {
                this.p = p; }
            public BigInteger getQ() {
                return q; }
            public void setQ(BigInteger q) {
                this.q = q; }
            @Override
            public String toString() {
                return "d=" + d +
                        ", p=" + p +
                        ", q=" + q +
                        '}'; }}
        static class OpenKey {
            private BigInteger n;
            private BigInteger e;
    public BigInteger getN() {
                return n;  }
            public void setN(BigInteger n) {
                this.n = n; }
    public BigInteger getE() {
                return e; }
     public void setE(BigInteger e) {
                this.e = e;}
    @Override
            public String toString() {
                return "n=" + n +
                        ", e=" + e +
                        '}';} }
```

```
public OpenKey getOpenKey() {
        return openKey; }
public PrivateKey getPrivateKey() {
        return privateKey;}
    @Override
    public String toString() {
        return "KeyPair{" +
                "openKey=" + openKey +
                "\nprivateKey=" + privateKey +
                '}';}}
  public static BigInteger
generatePseudoPrimeNumber(int
amountMinBit, int amountMaxBit) {
    int count = 0;
    BigInteger number = null;
    do {
        count++;
        System.out.println("Attempt number : "
+ count);
        boolean satisfact = false;
        while (!satisfact) {
            number =
generateRandomBigIntegerFromRange(new
BigInteger(String.valueOf(2)).pow(amountMinBi
t),
                    new
BigInteger(String.valueOf(2)).pow(amountMaxB
it));
            satisfact =
checkSimpleConstraints(number);
        }
        System.out.println("number: " +
number);
    } while
(!millerRabinProbabilityTest(number));
    System.out.println("result of Miller-Rabin
test for this number: " +
millerRabinProbabilityTest(number) + ", amount
of attempt to find = " + count);
    System.out.println("Pseudo prime
number:" + number);
    System.out.println();
    return number; }
  public static BigInteger
generateRandomBigIntegerFromRange(BigInteg
er min, BigInteger max) {
    BigDecimal minD = new BigDecimal(min);
    BigDecimal maxD = new BigDecimal(max);
    BigDecimal randomBigInteger =
minD.add(new
BigDecimal(Math.random()).multiply(maxD.subt
ract(minD.add(BigDecimal.valueOf(1)))));
    return new
BigInteger(String.valueOf(randomBigInteger.set
Scale(0, BigDecimal.ROUND_HALF_UP))); }
public static boolean
checkSimpleConstraints(BigInteger number) {
    boolean result = true;
    String text = String.valueOf(number);
    int length = text.length();
    int sum = 0;
    for (int i = 0; i < length; i++) {
      sum += Integer.valueOf(text.substring(i, i
+ 1));}
    if (text.substring(length - 1,
length).equals("0") ||
          text.substring(length - 1,
length).equals("2") ||
          text.substring(length - 1,
length).equals("4") ||
          text.substring(length - 1,
length).equals("5") ||
          text.substring(length - 1,
length).equals("6") ||
          text.substring(length - 1,
length).equals("8") ||
```

```
            sum % 3 == 0) {
        result = false;  }
    return result; }
  public static boolean
millerRabinProbabilityTest(BigInteger p) {
    int counter = 0;
    boolean changeCounter = false;
    int k = new Random().nextInt(46) + 5;
    //Крок 0
    BigInteger numberS1 =
p.subtract(BigInteger.valueOf(1));
    BigInteger d;
    int s = 0;
    while
(numberS1.mod(BigInteger.valueOf(2)).equals(n
ew BigInteger(String.valueOf(0)))) {
        s++;
        numberS1 =
numberS1.divide(BigInteger.valueOf(2));
    }
    d = numberS1;
    System.out.println("d = " + d + "\ns = " + s);
    while (counter < k) {
        changeCounter = false;
        //Крок 1
        BigInteger x =
generateRandomX(BigInteger.valueOf(2), p);
        BigInteger gcd = new
BigInteger(String.valueOf(x)).gcd(new
BigInteger(String.valueOf(p)));
        if (!gcd.equals(new
BigInteger(String.valueOf(1)))) {
            return false;
        }
        //Крок 2
        if (x.modPow(d, p).equals(new
BigInteger("1")) || x.modPow(d, p).equals(new
BigInteger("-1").mod(p))) {
            counter++;
            changeCounter = true;
        } else {
            for (int r = 1; r < s - 1; r++) {
                BigInteger xr =
x.modPow(d.subtract(new
BigInteger("2").pow(r)), p);
                if (xr.equals(new BigInteger("1"))) {
                    return false; }
                if (xr.equals(new BigInteger("-1")) ||
xr.equals(new BigInteger("-1").mod(p))) {
                    counter++;
                    changeCounter = true;
                    break; }}}
if (!changeCounter) {
  return false;} } return true; }
  public static BigInteger
generateRandomX(BigInteger minLimit,
BigInteger maxLimit) {
    BigInteger bigInteger =
maxLimit.subtract(minLimit);
    Random randNum = new Random();
    int len = maxLimit.bitLength();
    int randomBitLength = new
Random().nextInt(len) + 1;
    BigInteger res = new
BigInteger(randomBitLength, randNum);
    if (res.compareTo(minLimit) < 0)
        res = res.add(minLimit);
    if (res.compareTo(bigInteger) >= 0)
        res = res.mod(bigInteger).add(minLimit);
    return res;}}
```