Міністерство освіти і науки України НТУУ «Київський політехнічний інститут» Фізико-технічний інститут

КРИПТОГРАФІЯ

КОМП’ЮТЕРНИЙ ПРАКТИКУМ №3

Криптоаналіз афінної біграмної підстановки

Перевірили:
Чорний О. М.
Завадська Л. О.
Савчук М. М.

Виконали:
студенти 3 курсу ФТІ

Ракович Дарина ФБ-73
Пекарчук Данило ФБ-74

Київ – 2019

Мета роботи

Набуття навичок частотного аналізу на прикладі розкриття моноалфавітної підстановки; опанування прийомами роботи в модулярній арифметиці.

Порядок виконання роботи

0. Уважно прочитати методичні вказівки до виконання комп'ютерного практикуму.
1. Реалізувати підпрограми із необхідними математичними операціями: обчисленням оберненого елементу за модулем із використанням розширеного алгоритму Евкліда, розв'язуванням лінійних порівнянь. При розв'язуванні порівнянь потрібно коректно обробляти випадок із декількома розв'язками, повертаючи їх усі.
2. За допомогою програми обчислення частот біграм, яка написана в ході виконання комп'ютерного практикуму No1, знайти 5 найчастіших біграм запропонованого шифртексту (за варіантом).
3. Перебрати можливі варіанти співставлення частих біграм мови та частих біграм шифртексту (розглядаючи пари біграм із п'яти найчастіших). Для кожного співставлення знайти можливі кандидати на ключ (a,b) шляхом розв'язання системи (1).
4. Для кожного кандидата на ключ дешифрувати шифртекст. Якщо шифртекст не є змістовним текстом російською мовою, відкинути цього кандидата.
5. Повторювати дії 3-4 доти, доки дешифрований текст не буде змістовним.

Критерій відобру ключів:

Використовували неможливі біграми для відсіювання неможливих результатів.
Використовували такі неможливі біграми: 'аь', 'эь', 'еь', 'йь', 'уь', 'оь', 'ыь', 'яь'.

Таблиця біграм:

З шифртексту:	Теоретичні:
сг	**ст**
жэ	**но**
нг	**то**
ям	**на**
цр	**ен**

Зашифрований текст:

юммутмкйсйумтцбишчоцхйнкхйяхклзкугргтвдньлмгсбтмейашрэлшогэгклсмтцзлжфбтдлычтфыляунгфищйэзргччсбьцжм
ьижнулхщьлкюэклксыямямбйжцтпогсбэищмзмхшсчмддуилойэйугюцдтруцвдуампзэйбуззцюжнвкбхгвргфбчишчжпэгкнрш
зццлбгвптмвннггшргэмбхогрирумчилцнвцвпжэбтцтвпеэлкжэзйуццлкшцбоцюццнзлмчяуфбяцтбжэеэсйлкдмеццатмбцца
ймумгхьцнгюцьцдхшзеэпнсбжэящбгилнгтзяунивпеэсмямешуйэйшйэсозгкшйяментэьхрхюзгкхййзгкнйфцгзбйаыыцрхрг
ычксдяьцзллтлгтмбхьэецшзюхщйххшзтбьцтмтэеэьхрхжэпмтэсмжжбйаыыцкирийзсмеивфзэсглйллжмцкэсткяжюццтдлкш
укикейяржэййнгзлахрхмйщмймммнзиьцтмнипйхуфуббьлвфоцлйямярсмюмгчжюфбфмтмжжжэекчэзмхйейаклккддуилридйщй
укнйриеэцукссмтцбгопямржшйцлритбмкэбьцогфгебээяждивфбшквдусгбгвццнтчойцкщбдярхтфьлжнммжэцксмхуоцяффу
ббкияйэншцфуклщмяжллфгргяэувшчбубиьлцнммейцюклримвнйяэзвьлыщебрвщйзиычелцнфихйщнзимйтээцяфебнгтммйсг
гнзиьцтмугклемшйхбщюржгзыцлдргзлафжтзэфкцрагахрхтбилдтвпшчейоэбкагебэеыццнеэдьэюцоцрхюцфщшйдшрцшмжэ
жэшушмямшиьлйзюцтмыццюгэдшрцтввфцтргогкнзкдуешюказдмзипйбшрэзпямршгэвнтэикцрхктмеэыгыцхчфужмсчмйтцюх
днейщйжнсмвксйбхмйвпхкгнлкймвчбгфипйагригйсгнийзхкхпзлриэхюмишсэнгцлатшйэнзкбхиянггэгзагмкгнцргпцнцр
гнвпыцогфгрхксдктчксгйуэцкгшбшрэзммпямпшбхзаопзлриэхгвумэмулыцщюкклктмнгэфчэилмйщмошяэнгфйзнюццкйэжм
ьиычвпяйэнлиюмсгзлоиястмгржэсчклнгвушеьйщмнчыфэгдэомбжбипйзэщцргдмпбэеыцфуцнейошгпцнцрэхрхтэрэюксйбх
шзечявыцеэймлкугжтумзйжэдмчйоцоияптмумцуяйэнклркжтзэгкьэвнсбычниймззтмпчцнзиычщгюцшчцнекмнбткинйдйшх
цлкектжкрмвчгнгктэгшщбэеыцявтнцрсбфмзингогжэагргдмсчжныййжибщчпшсгммзингогкйбрщмумпмывтнцрргкйгкояьц
ьлыццношэйицфбкляэьэнкпзузпмьнцктцрхтпзлйдмйсбкляэцкцюриычилизяцлйвньхмдллринйргчяцшэмяхдэсчахыццлщм
змхшдншшфэяхрхсиуьнгкйейыилшсгцалксгкйэмлшсйщмдздэшйдэржчэнюриычилюцпйфйсыямгезлгнблднцрниямбйдшбйоу
зззиойибэеыцыпшчяфбкдшршкйзэоушмпшсгзлгвтнбжэхтпзлыцпвдуикрцтвриднлкмгсбтмейхйбшшхкшйждццлйздмяцемэв
впэисгдмчэтыршцлатгшруоиилйзуарвумяццлйзюмнйоээцсбтмщмсчдншшфэеньхпйфйрвюцелргггшгэюцбирзьэбрдтимдшлл
жеьлшйугщюычтфпмомнйнииятмьцфязлрвьцмйжгхщилкезлгнлбклдуцалксгжжеэцуращмеццнршсэгуьцдхщюгэюцдмзгрхсч
жнмвьлйзхягэтыямщмшимхрхргычравфжэяэбпямейдйлрямдянйямжэяэзмбрюмжэсгизэйошюцюцтпыцьлжнфиргмгрхойэгум
смхурхямцкашзрафпмэмяэящниилчюжнейжвнйэзицюхлжцклкщцмхжжамукэцьлдязагшюмхйщмахтмбкгйтэйлапгнхкьцчякк
увхщцхешоэафкйтзюмбшексгцанйфцуатчбипвчкнйрцимфйнэлклйюмечшзтццннишчжэдмсгахжэяюжэдэсэпмвклжчэлипйгй
крекйкгйкрекэхчэгчбгщхжнилеэшиюцсмсгщйсмтэьэафюмецчяуавчыфнэлкщмзмыхюхнйкйзэщйзхмйрмзмбжйкбххждщзлни
екквжннйзтычшмбшчэмпюхриошзэыррвьлвпхкеэгйшхжмсгбгхксчмезмщмеццютвзэмутмчэцкэпнгкйыгщюмкплгмаюычзлий
тшгкдкюхтнвпхктчэгоцщюьэщцойдэшйщнбльлйзсчодьэецрапксграюмямпнвптмхфзльццатчбиюубушечйдяпвщйнntэжлел
жндхрхюцжьэхвпнгцююущюжмвлиягэнгзэвюычэпьцтмзлхщюцгвклтзнгтммучкафкйбцобмчфббгфбщцриачдуоизйщмйзнйтэ
жмфйкэнггэяжвннипйсччяоисбтьфщзлращмчизлюхойэшпмсмнкяхкюриычилпйфйхйбшшхрхрхцаагдуичрхжжбшрэбпямтмжж
нюдужэпкжэнгйздйыищюизьимчшцфужтдлтвжнтэикцрпмрмйцидеэзэцлогачжнзгзэжжжэбкйлжмжэвлдяяйхкйлыццаээщюза
мклкзарижкеэлфебнглгугимжэймдчзлфуззпйтмсмээмгжнебойэгумрбгнямрйвйыишчсмаооцюэчкечюхрипшсгвцтпыцяйэн
лизлюхшзашйцшзнкшоюцжзсмнкзмпнщйдэььмдсифямйтчбичцоцемозфцидшшщмжжкжумсгычядмгргбуыцсгкйицрахйкрхюду
мхргычраюмжэсгизэйошжцебьцьйэнжтцажэяэзмэспзтцфужтатумээцргчжюауешнэшйсмоонггкыцгншчсэьгыцгвчкщмчецк
сифягнсмзйиццатчбиюубушееэмбщхдшбпямямдяомяртмпмшюксоказмймокбчтфкйписгьлюмпмзйяхахдшоианкццлйзжлцнсм
жжщнклшйбцрасингтэхкгнцркмщйсгшчтмжжляилфщэмзмпмнютфнгпкйхрхзэкфрхоцпкчэнгузлужтумцштчфбднейснюмгуыц
вгрхлйамумзйкрсмткйлшямчфбечюхдшгубушееэоибхычцшбйнкхйпицквшчшалнгьлсччюргринйицравэашшхцадэмгзэхещч
ахойахтмржзмзэбххжямхядяоэкрэхжнилхкллритмюмщвниьлвфбитчрхфмкыямухжнилхкжэбшзйсгоцриогйзувахцашйяфеб
нгилхйбшшхбгкгклдхтвжншшфэдгпвшчюшрцбуклщмжэгйгншшфэцрргвфчэьэбрвэашшхэкюмсгбиилоццнямшюычгшвеьцапфж
влпвйеэхткйлкмщйсгшчюмпйнгпйщмецшцрирайфмыгтчбиийьцьцфужтрунгсмнкцршйюмтцехрхтээмжэжэтжчэнгйзтмюцрхрх
жжзэсгнгбхклдхюцдятмтэпнебэкжнтэтмикэнзкомярзйшйзэсгнгхйбшшхцнафббнйцксбтнтэикенщцияьцрурикливцщюшшсэ
зеэгеэбхтфьлюцжэсмшйщкукюмхбргчкээьтилгоычгнейеюриычогагвфсбнйжэюмярвшбхтфьлцтцтшееэюцдяпвчккйжцриий
пйгйжэюцднзмбйфждэпмэмеирэнйкрхкчхычфккхяддшрцризэжцидгйсгяджжхжцрпбхкафцнмвбгфгруцнсбьцрхсмвщжюдуоц
ришзтчбикшжрнгвнлнтэешскямхуебчяйецвумгэчэцрсиюссмнкяхфмжэейфйомярзйшйюмжэпмткэняубушееэймыиьлйзймсг
цлэкбкщмеиансмпкпидудмстилгоычгнейумсгоцриилсмнкяхшкзбичбнммэнклвщцнцлатсмжвдумезлзцсбэиргтэщгжнеэдэ
змьисйзэгшвкэцриийпйгйжэгэчуббосдэпйхиебшшоэухявахшзсижтдуьцебьцнввпсбкцгнсчжнилилдязаеэямвннгрхтбиз
дэечапьлхйбшшхшзбшрэпиатрунгбуриклвцобднейснюмгуыцпцапцюизшйпмдмнйэвахияогсбшйагопшчямхштчбифмцксгиз
лкдмтлсбьльцдхмдсйбжщйсшщмярфжчхычфктмпнтмдмэмецобднсбьцебвубушееэюмпмдчогзлвцтвргщхьлдудшафцкхцжнсм
ьтэерэеэюмэмсбаыямээрэгмржзцриьцжэлгзэчэбиушкйщйфжугрицншйжэбшлклиеэлккмйцккикхйяхрхюцмчцнейзиргжэсг
клдхрихчцнейгквнлйллогклдхимжэсчхяоингйзюфхйццсбултббгъцебдяьццтрувьфащмецпвчкхйэньхиябшгцбузлчвдусг
чкнэцршшоэухэиямегмкфйгэсиэетбюмсмнйцксббхтфьлхйбшшхебшйсгюутмдэчэзтильэтьфяичжэлксгжжщмямжжкцапямяк
црвняпямгнцржэсинглдычмецюргклдхжткгзэгкцрюмгйкрзйччсбьцрвфбьцренгешгэдмшпцюриюужтоияйейбрюцшйюмугог
злвцтвдшяхычфккхкщщюцадмйвчяоинггнэмнгщюгэюмибнгяпямгйяхмйзэпмсгюхрубиияшйэхешуцидзэтмэгумзвригншйсч
щюцллгкннвйесмтцнисчтбшчшшпеяпсбезгкцрдмпйфрхййзлбилфиэээмрчжнсмццфббиямэмткнйтфжэнгпввпсбявргфбопям
внниахешчтьэщмьдяйзипйзэсгнгхщкйейцхщюгэвуйецвйесмгйсгфищйэзщюззхйцрйздмечжюдэгйгкцрээошщбьлксечцнрг
фзюмцкрэлнуаяэсиебамсйэнгмиибгогфипйэчтфшкагюцбишчтмгнейсгкйицуыямсицасгдмэмзмпмсгыццютфнгпкгшжэпмбб
нгфивунгркгйяхфаиктмнчфуоцнгклдхимнчфбфжшэстофкиебюмгэчэлщзлрабхзлнгцнгйтзпмдмяртмдннйгэсэярыйюпямгг
чяпвюцлгтмпйхпцнгйзмпкеэмшэнлиеббхопфжбктмэмбйзимвумсгойэйшйжцяйейтбыйфжшйбблйейжэамукцрщцмдыхтнгэди
ебдмэсжжржафгэлкдэсмтмашсгзэшйцкоипймбзлхкыузлвнлдгйзмсйсцргщюгэчэтцяпцюжтдлжтюузлбгфидйщйхжкйгйцутт
клретбвузлебкксчкезмбрржпмднсгявнйцкнвчкбшяжчэцрпйфйдэтзгкяпямржшймдгднбббгрхдяшшпепвщшпклбрржсмяп
сбжшэээмжвчкумсжюмнтччюбгфгзэхшоимвлкибгкнычтзхкямбйдужэафзэпмсгуаякжцднчцахткйлкмщйсгшчьэбхрищшйж
дццлйзсгмчюхофжэзфжэлнеэдзгубушееэбнтэтмжллнкнлкшхфбомплгкжжеиякжцыцюхщююцгншшфэентэхкошпмдмярсгиксб
ямхугвумпмцятбнйбщысгшсгнкщйцкшшпеемщдвууажмбшчэмпюхогчяйксйыгтвклодгввпсбругвргфбопямюцмдргвцьцфукл
щмргжлэхосшпзлриэхкшдмпзейщмргэсхйакщмумгэдэсгбиззхгюмюмейцроивфччжьосэншйфйяцтфсбэсчштхщюгэбжкйбррш
гкцкщгргттьцсбйеяйедргвцьцфуклщмжнтэйекншцяйукшйзнебтбймомакжэпкжэйеиуббббгммукртяфюмсгычюцшйжвфбхьюх
щюгэсмсэйекнлиээбшэээмнищйэзюмокзмйминзирибгвцюхямжтагфбшйпмнэсжчэнгйзцкщгргттшцемциыцййфцюхямыцяфжэ
нгцксйщмбгантэтмикшрйздэсчбибхтбокзмймкмиыгэхкжвьэямрнщйуквкюцриймечбгсмямхжщнебпйфйжжзэькчэнгщкгшрш
ямцхямтэькфццнвкшйюмсиойюлдзяачйгйбмзцбивфцнумецщэблцнжшсмямумщмжэаляпядччжьяйэнэфсбтмаемчфытсщлувкл
укдюхнщизаснеэдзэмецияєжсмсээфсбєюмсмсэщнфшпйозаушенишчгкейзмсэсгсыямухжнилхктклвушмумнкзицнякикей
яржэбнтчфзкмьгоцостмсгилэхщкгмсммпьцрхумгйямщмафщйзхшмсйкрюмярфкейяртмргкйгэцрцйнйцкдуэиоцпксмчэжжгэ
щцойлкцйнйкрдуэиоцпксмщцкшякрнлчшйашхйошямбцлгктиляхоидшлкнйщмкрсмсгмхдшетлкоэаекссмнфыцгштмбхфбвэнб

рхбгрхдмфццнсйкыямдгзэжжййэйшйвнниеббхтфьлюцмчцнейзгрхлшсгвпямжэкййцьхтфкюшшекгкулнишчжмбшчэмпюхчюжн
юмукюмбкшшхфжэпкршбйжэашбшькнюдуяжжнтэшйьнбтяфикжэафдуэиоцпксмикцкдутмжвсййжжэслдязасмчубимвумсгжнбб
цнмелкюмллшезаейбхклдхриыфгвьэгклкдэагднсбьцфукльэлхычфккхшзгшвкгугнлгрершщиебовдумчаххйбшшхгвкляфых
мчрэьикгжьилсгмчфбдуфжчулкицрасмзйнйсмэнейпццлпйщйвчшцксдэсгфягшккшинйщгчкашрэтхрвтвлгиукльэумвтилпи
кгклдхвпямймкйчеилрвщйжжщмецксеузлгшдззэомтэлкюзцрпмйлмчриафдмярейпйхпцнбйргсчбиогфиясагопшчгнэшювум
дэапгнгшхгзэхшдмччкюжэюмэмухтнзмдмфхруядргогнгклдхтвахычшйаепвумтшшйпйдглйцшгйтздмчиычезжцосэицнээяж
пводчэярфжьцебшчьцжзтмлипйтэхкдябужнгквигншбгввпейицщюривфзэсгэкхбмкбкбйьэцкпбыццющйюнпктэцрйзовшчхш
нмсйщйзрмэ

Розшифрований текст для ключів 300, 400:

поздновечеромнаверандесиделколяичтотописалвтемнотебумагуитутолкомнельзябылоразглядетьвремяотвре
менионвосклицалагаилииэтотожезначитемувголовуприходилоещечтонибудьподходящеедляегоспискапотомд
верьчутьстукнулаточновсеткуотмоскитовудариласьночнаябабочкалинашепнулауфманонаселярядомснимнак
ачеливоднойночнойсорочкенетоненькаякаксемнадцатилетняядевочкакоторуюещенелюбятинетолстаякакпяти
десятилетняяженщинакоторуюуженелюбятноскладнаяикрепкаяименнотакаякакнадотаковыженщинывовсяко
мвозрастееслионилюбимыонабылаудивительнаяеетелокакиегособственноевсегдадумалозанеетолькоподруго
муоновынашивалодетейиливходиловпередилеовкаждуюкомнатучтобынеуловимоизменитьтамсамыйвоздухп
одстатьнастроениюмужаказалосьонаникогданезадумываетсянадолгомысльтотчаспередаваласьотееголовыпле
чампальцамипретворяласьвдействиетакнезаметноиестественночтолеонесмогбыдаинехотелизобразитьэтокаки
милибочертежамиэтамашинасказалаонанаконецненужнаонанамдаотозвалсяонноиноганужнопозаботитьсяио
другихявотвседумаючтотудавставитькинокартинырадиоприемникистереоскопическиеочкиеслисобратьвсеэт
овместевсякийчеловекпощупаетулыбнетсяискажетдадаэтоиестьсчастьесочинитьтакуюхитруюмеханикудума
лончтопускайучеловекапромоклиногиилиноетязваилиегомучаетбессонницаионворочаетсявпостеливсюночьн
апролетидушуегогрызутзаботыавсеравнотвоямашинадастемусчастьекактамагическаякрупинкасоличтоброше
навокеанивечнорождаетсольиобратилавсеморевсолянойрастворктонерасшибсябывлепешкулишьбыизобрести
такуюмашинупустьемуответитнаэтотвопросцелыймирпустьответитвесьгородокпустьответитженалинасмуще
нномолчаласидярядомснимнакачеляхиеемолчаниеговорилояснеевсякихсловлеотожеумолкзапрокинулголову
ислушалкаксвищетветервгустойлиствемогучеговязанезабывайговорилонсебеиэтотшелестлистьевтоженужен
длятвоеймашинычерезминутуверандаопустелапустыекачелинеподвижноповисливтемнотедедушкаулыбнулся
воснеонпочувствовалэтуулыбкуудивилсяейипроснулсяполежалнемногоприслушалсяксебеипонялоткудаонав
зяласьибоонуслышалнечтогораздоболееважноенежелипениептицилишелестмолодойлиствыкаждыйгоднаступ
алденькогдаонвоттакпросыпалсяиждалэтогозвукакоторыйозначалчтотеперьтоужлетоначалосьпонастоящему
ононачиналосьвотвтакоеутрокогдактонибудьиздомочадцевилигостейплемянниксыниливнуквыходилналужай
куподегоокномиметаллическиеножииспицыкружаизвеняподушистойлетнейтравеприлежнообегалиеепокраям
насевернавостокнаюгназападописываявсеменьшиеименьшиеквадратыкосилказвонкострекоталаизподножейб
рызгалиголовкиклевераредкиезолотыеискрыуцелевшихпослесбораодуванчиковмуравьипалочкикамешкиоста
ткипрошлогоднегопразднованиячетвертогоиюляобгорелыешутихиикусочкитрутаноглавноезанейстлалсяпрох
ладныйчистыйпотоксочнойзеленойтравыдедушкеужепредставлялоськаконащекочетегоногиохлаждаетразгор
яченноелицонаполняетноздриизвечнымароматомвновьродившегосялетаиобещаетдамывсевсепроживемещеце
лыйгодвеликоечудокосилкаговорилсебедедушкакакойэтодураквыдумалчтоновыйгодначинаетсяпервогоянвар
янадобылопоставитьдозорныхкараулитьросттравынамиллионахлужаекиллинойсаогайоилиайовыикказаметят
чтоонасозреладлясенокосавтосамоеутровместофейерверковфанфарикриковпустьначинаетсявеликаябурнаяси
мфониякосилоксрезающихсвежиетравынанеобятныхлуговыхпросторахвтотединственныйденьвгодукоторый
понастоящемузнаменуетсобойначалолюдямнадобыбросатьдругвдруганеконфеттиинесерпантинапригоршнис
вежескошеннойтравыдедушкахмыкнулчтотоужбольнодолгуюфилософиюразвелвсталподошелкокнуивысуну
лсявласковыйсолнечныйсветтакиестьфорестерновыйжилецмолодойгазетчиккакраззаканчиваетряддоброеутр
омистерсполдингтакеехорошенькобиллсжаромкрикнулдедушкаивскореужесиделвнизуиуплеталприготовлен
ныйбабушкойзавтракширокоеокнобылораскрытоижужжаньекосилкисловноподпевалозавтракуотэтойкосилки
надушестановитсяспокойнеезаметилдедушкатытолькопослушайтеперьужнедолгонамееслушатьотозваласьба
бушкаипоставиланастолгоркупшеничныхлепешекбиллфорестерпосеетсегодняновыйсорттравыеененадобудет
коситьнепомнюкактамонаназываетсяноонакаквырастетсколькоонужнотаксамаиостановитсяибольшенерастетд
едушкасизумлениемуставилсянаженудовольноглупаяшуткасказалоннаконецидипосмотрисамбиллфорестерго
воритэтоземленапользусказалабабушкаонужепривезновыесеменаонисложеныздомомвмаленькихкорзинкахн
ужновразныхместахвырытьямкиизасыпатьтудасеменакконцугодановаятраваубьетвсюстаруюитогдаможешьп
родаватьсвоюкосилкуонатебебольшенепонадобитсядедушкасорвалсясостулаимигомвыскочилводворбиллфор
естеростановилкосилкуижмурясьотсолнцасулыбкойподошелкнемувоттактосказалонвчеракупилновыесемена
дайдумаюзасеювамлужайкупокаясвободенаменяпочемунеспросилилужайкатовсетакимоязакричалдедушкаяд
умалвыбудетедовольнымистерсполдингничегоянедоволенпокажитемнеэтучертовутравуонистояливозлемале
нькихчетырехугольныхкорзиноксновомодныимисеменамидедушкаподозрительнопотыкалоднуизнихноскомба
шмакапомоемуэтосамаяобыкновеннаятраваавыуверенычтовасненадулиявкалифорниивиделкаконарастетвотн
астольковырастетивсееслитолькоонаприживетсявздешнемклиматенамуженабудущийгоднепридетсякаждуюн
еделюподстригатьлужайкувтомтоибедасвашимпоколениемсказалдедушкамнестыднозавасбиллаещежурналис
твыготовыуничтожитьвсечтоестьнасветехорошеготолькобытратитьпоменьшевременипоменьшетрудавотчего
выдобиваетесьоннепочтительнопнулкорзинкуногойвотпоживетесмоетогдапойметечтомелкиерадостикудаваж
неекрупныхраноутромповеснепрогулятьсяпешкомневпримерлучшечемкатитьвосемьдесятмильвсамороско
шномавтомобилеазнаетепочемупотомучтовсевокруглаготухаетвсерастетицвететкогдаидешьпешкоместьврем
яоглядетьсявокругзаметитьсамуюмалуюкрасотуяпонимаюсейчасвамхочетсяохватитьвсесразуиэтонаверноест

ественноэтосвойствомолодостиногазетчикунадоуметьвидетьимелкийвиноградanетолькоогромныеарбузывам
подавайцелыйскелетасменядовольноисследапальцевчтожтожепонятносейчасмелочикажутсявамскучнымином
ожетвыпростоещенезнаетеимценынеумеетенаходитьвнихвкусдайвамволювыбыиздализаконобустраненииивсе
хмелкихделвсехмелочейнотогдавамнечегобылобыделатьвперерывемеждубольшимиделамиипришлосьбыдои
сступлениянридумыватьсебезанятиечтобынесойтисуматакужлучшепоучилисьбыкоечемуусамойприродыподс
тригатьтравуивыпалыватьсорнякитожеоднаизрадостейжизнисынокбиллфорестерласковоулыбнулсястарикуз
наюзнаюсказалдедушкаястановлюсьслишкомболтливымвжизниникогонеслушалстакимудовольствиемтогдап
родолжимлекциюкустсиренилучшеорхидейиодуванчикитожеичертополохапочемудапотомучтоонихотьненад
олгоотвлекаютчеловекауводятегоотлюдейигородазаставляютпопотетьивозвращаютснебесназемлюиужкогдат
ывесьтутиниктотебенемешаетхотьненадолгоостаешьсянаединессамимсобойиначинаешьдуматьодинбезпосто
роннейпомощикогдакопаешьсявсадусамоевремяпофилософствоватьниктообэтомнедогадываетсяниктотебяне
обвиняетниктоинезнаетничегоатыстановишьсязаправскимфилософомэдакийплатонсредипионовсократкотор
ыйсамсебевыращиваетцикутутоткотащитнаспинепосвоейлужайкемешокнавозасродниатласуукоторогонапле
чахвращаетсяземнойшарсэмюэлсполдингэсквайрсказалоднаждыкопаяземлюпокопайсяусебявдушевертитело
пастиэтойкосилкибиллидаороситвасживительнаяструяфонтанаюностилекцияоконченакрометогоизредкаочен
ьпользительноотведатьзелениодуванчиковавыдавноелизеленьодуванчиковнаужинсэрнебудемуточнятьбиллк
ивнулилегонькостукнулближайшуюкорзинкуноскомбашмакатаквотнасчетэтойтравыяещеневсевамсказалона
растеттакгусточтонаверняказаглушитиклевериодуванчикигосподипомилуйзначитуженабудущийгодмыостан
емсябезвинаизодуванчиковиниоднойпчелынадлужайкойдавыпростосумасошлипослушайтескольковызаплати
лизаэтисеменадолларкорзинкаякупилдесятьштуквамвподарокдедушкаполезвкарманвытащилстаромодныйдл
инныйкошелекотстегнулсеребряную застежкуиизвлектрибумажкипопятьдолларовбиллвытолькочтосовершил
ипревыгоднуюсделкузаработалипятьдолларовизвольтесейчасжеотправитьвсюэтучересчурпрозаическуютрав
увоврагнапомойкусловомкудахотитетолькопокорнейшепрошунесейтееуменяводвореязнаюувассамыепохвал
ьныенамеренияноявсетакиужедостигвесьмапочтенноговозрастаисмоимижеланияминегрехсчитатьсявпервую
очередьаа.

Код програми:

```
from collections import defaultdict, Counter

rus = ['а', 'б', 'в', 'г', 'д', 'е', 'ж',
        'з', 'и', 'й', 'к', 'л', 'м', 'н',
        'о', 'п', 'р', 'с', 'т', 'у', 'ф',
        'х', 'ц', 'ч', 'ш', 'щ', 'ь', 'ы',
        'э', 'ю', 'я']

def bigram_in_int(bigram):
        global rus

        return rus.index(bigram[0]) * 31 + rus.index(bigram[1])

def int_to_bigram(number):
        global rus

        return rus[number // len(rus)] + rus[number % len(rus)]

def gcd(a, b):
        '''
        recoursive gcd alg
        '''
        if b == 0:
        return a
        else:
        return gcd(b, a % b)


def inv(a, m):
        a = a % m
        for x in range(1, m):
        if ((a * x) % m == 1):
        return x
        # if no results:
        return None
```

```
def linear_eq_alg(a, b, n):
	'''
	solves a * x = b % n for x
	'''
	gcd_res = gcd(a, n)
	if gcd_res == 1:  # relative prime
	inv_res = inv(a, n)
	if inv_res is None:
	return None

	x = (b * inv_res) % n
	return [x]  # list so it works properly for >1 case

	elif gcd_res > 1:
	if b % gcd_res != 0:
	return None

	if b % gcd_res == 0:
	res = list()
	div_a = a // gcd_res
	div_b = b // gcd_res
	div_n = n // gcd_res

	if inv(div_a, div_n) is None:
	return None

	x_0 = (div_b * inv(div_a, div_n)) % div_n
	for i in range(gcd_res):
	res.append(x_0 + i * div_n)

	return res


def validate_deciphered_text(input_bigrams):
	'''
	looks for impossible bigrams
	'''
	impossible_bigrams = [  # TODO: add more
	'аь', 'эь',
	'еь', 'йь',
	'уь', 'оь',
	'ыь', 'яь',
	]

	for bigram in impossible_bigrams:
	if bigram in input_bigrams:
	return f'Impossible bigram found: {bigram}'

	return None


def get_bigram(in_list):
	'''
	mixes and returns all bigrams
	'''
	result = list()

	for first in range(len(in_list)):
	for second in range(len(in_list)):
	if first != second:
	result.append((
		in_list[first],
		in_list[second]
	))

	return result
```

```
def get_key(theor_bigram_pair, encr_bigram_pair):
        '''
        returns key
        '''
        x_1 = bigram_in_int(theor_bigram_pair[0])
        y_1 = bigram_in_int(encr_bigram_pair[0])
        x_2 = bigram_in_int(theor_bigram_pair[1])
        y_2 = bigram_in_int(encr_bigram_pair[1])

        linear_eq_res = linear_eq_alg(
        (x_1 - x_2),
        (y_1 - y_2),
        961  # m ** 2
        )
        if linear_eq_res is None:
        return None

        key_arr = list()
        for a in linear_eq_res:
        b = ((y_1 - a * x_1) % 961)
        key_arr.append((a, b))

        res_key = list()
        for key in key_arr:
        a = int_to_bigram(key[0])
        b = int_to_bigram(key[1])
        res_key.append((a, b))

        return res_key


def decipher_cipher(text, key):
        a = bigram_in_int(key[0])
        b = bigram_in_int(key[1])

        inv_a = inv(a, 961)

        if inv_a is None:
        return None

        res = ''
        for index in range(0, len(text)-1, 2):
        y = bigram_in_int(text[index:index+2])
        x = ((y - b) * inv_a) % 961
        res += int_to_bigram(x)

        return res

def crack_this_affine(bigrams_theory, bigrams_prac, input_text):
        logs = str()
        common_bigrams = get_bigram(bigrams_theory)
        encrypted_bigrams = get_bigram(bigrams_prac)

        logs += 'Попарно перебрані пари найчастіших теоретичних біграм:\n\n'
        logs += str(common_bigrams)
        logs += '\n\nПопарно перебрані пари найчастіших біграм з шифртексту:\n\n'
        logs += str(encrypted_bigrams)

        keys = set()
        for common_bigram_index in range(len(common_bigrams)):
        for encrypted_bigram_index in range(len(encrypted_bigrams)):
        mixes_keys = get_key(
        common_bigrams[common_bigram_index],
        encrypted_bigrams[encrypted_bigram_index]
        )
        if mixes_keys is None:
```

```
            continue

            for key in mixes_keys:
            keys.add(key)

            keys = list(keys)  # moving from set to list so it's eathier to work with

            logs += '\n\nКлючі що дають неможливий текст: \n'
            matched_texts = dict()
            for key in keys:
            deciphered_text = decipher_cipher(input_text, key)
            if deciphered_text is None:
            continue

            validation_errors = validate_deciphered_text(deciphered_text)

            if validation_errors is None:
            matched_texts[key] = deciphered_text

            else:
            logs += f'\nKey: {key} {bigram_in_int(key[0])} {bigram_in_int(key[1])}\n'
            logs += f'Text is not valid: {validation_errors}\n'

            logs += '\n Texts that passed validation:\n'

            for key in matched_texts:
            logs += f'\nKey: {key} or ({bigram_in_int(key[0])}, {bigram_in_int(key[1])})\n\n'
            logs += matched_texts[key]

            return logs

def find_most_frequent_bigrams(text):
            res = dict()
            for index in range(len(text) - 1):
            bigram = text[index:index+2]
            try:
            res[bigram] += 1
            except:
            res[bigram] = 1

            return [a[0] for a in Counter(res).most_common(5)]

def main():
            input_text = ''
            most_common_bigrams_theory = ['ст', 'но', 'то', 'на', 'ен']
            most_common_bigrams_practice = list()

            with open('text.txt', 'r') as f:
            input_text = ''.join([a.strip() for a in f.read().split()])

            # getting five most common bigrams from text:
            most_common_bigrams_practice = find_most_frequent_bigrams(input_text)

            logs = crack_this_affine(
            most_common_bigrams_theory,
            most_common_bigrams_practice,
            input_text
            )

            # TODO: write logs to file
            print(logs)
            with open('result.txt', 'w') as f:
            f.write(logs)


if __name__ == '__main__':
            main()
```

Висновок:

Отже, в ході практикума ми засвоїли принцип криптоаналізу шифру афінної біграмної підстановки, набули практичних навичок у частотному аналізі на прикладі розкриття моноалфавітної підстановки та опанували прийоми роботи в модулярній арифметиці.