Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Фізико-технічний інститут

ЛАБОРАТОРНА РОБОТА №3

З дисципліни «Криптографія»

«Криптоаналіз афінної біграмної підстановки»

Виконали:
студенти 3 курсу ФТІ
групи ФБ-73
Деркач Вячеслав
Михалко Дмитро

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

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 №1, 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) ,(b a 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.

Хід роботи:

1)Прочитали методичні вказівки до виконання лабораторної роботи

2)Написали програму для обчислення оберненого елемента за модулем з використанням розширеного алгоритму Евкліда

3) Знайшли ентропії монограм в своєму ШТ та знайшли необхідний(8 варіант)

4)Для кожного кандидата на ключ дешифруєм ШТ, перевіряючи його частоти букв, що повинні підходити середнім в мові

6) Розшифрували шифртекст за варіантом 1

Критерій відбору ключів:

Проаналізувавши результати 1 лабораторної роботи, ми порівняли ентропії монограм з ентопією з 1 лабораторної роботи.
Найближчі:
[944, 824]
4.6685610795705745

[944, 570]
4.663266842320915

[944, 111]
4.6756793453435535

Розшифрування тексту за варіантом 8:

Використавши основи модулярної арифметики, критерій відбору ключів та частотний аналіз, було знайдено таку пару ключів: [17, 94]

['жц', 'дэ', 'цэ', 'сц', 'оц'] Біграми шт

4.496611671406175 Ентропія

Вариант 8

Зашифрований текст

хбтйьнцнюбцвэтйвшлпнрклщяуйычвшлоыезбвацлнэдйвтыяввэшлньзгишньжддэйфжцзбнцребюаддлсучщмшюшрвдлцнжбйэ юпувукешщейужчвеяшцвыгжэьсаерврлйоцбилтдтщыыыбюеяшюрзбсцащьтшдйшеетюйояцэцсощолшякфнплыдюржоыыйврдтю илжшьпюнялишбркесшющмлрвдджшоещхвлявугиржкрцбвдэацлнжцюевлзшонхщюдтюильлтбюбжзэзюжтбфоцэпндпйштщвэрд нцлнэлщцтйщццэйпахбщчьариртйфьсцуувэанцйзблныиьзявшждэирчигцнхщэщэьттдювжыосчьбрсдэзуигццээяжддэюрэц уевлтбюбнйфцгеупзшддкефржцэшйшюбчшцбдэйузожбмббущфшэешэцуожбуффрьвдхбщьзоцгщмлбшлнвцлнзбэщйвчиявхьбу ижцньзсдкеьнщцкнчедлиннетыоищхюшхдюдйвхвунчндуижщцкннцнйчшсцащьтлщжыинлцыйсншесшкноебятшдлпзыксзйвюп ьсвэьщжэбяьцлцребяецлцгнцэбясцжцылщхюшжшяцлшуонрлигщэзкнсомиюшцлсцишгецвжоыруыддгшбшдтжлзщяягьцэюшен тшдлуутхщццэшлсожзштйфчесдвэнциэирхигщжбжцмрэяцэжожрэясцжццдпшышсщэшштхвтеэшсццшмбмиыкбвштсдювцешеяш эоужцежлюртщарыбфещхыклщщцяцчшюпеббутзйвчигщьтгцнннйфечвчивефныбдвхврдьнцэяшяоньйрунжиьзфшлфшизнаерд жлзщэниныймпйбщнзбсцйроссцьвдлцнжбтдчоющиргищхюшфбкноцтыьзсдкеьнщцплйуяшчэшллцжбжцюеявдэаеупмьфшкноц тыьзсдкеьнщцоиэяхбоеьштыиршибщвеоцыдяоньйрхвдээнбшмшкйщцщзезсхдбрызбмиаемлжцячбууызбрфжцаерннцнфжцае юббшщньцннзиыыбужшаэштюдсздэпнжчючмэсучщаесцчпжшышьзоцзбрццбцэыгсдпшяцаэацсоеейвцбцэыгамжцяцвлштвеоц яцреузчпнтбщкцуожбуффрщджцошюбэосцаннйшлдсыблннйливэяузбцэкедэябщцдыхулнспидлщтруялшхйщцщзддйвшэвоцэ эяфеыхшлявугноошйбнегуйбнеаефржцйшкнлшинэцошятжэьмжцборцсцюбгьлцыйьсбщжжштяшяцфещдддхияулннйнциэщхйб аеыхацдршибщшэишчйчэявдэаегиюшчшоештздэжшжцэеюкйбоеисболшснзбрцплянэлцшюшейрхлшжцвэлцрннйпвьльтюфжц гьсцувыдуюыцнфмлщцщзддазцуешцйтщштыбнйюпзбуьсцыроснцмвжкхьклцуыбцнтщвэшещэущхрфнябющузюбмпрцхттзщэцэ йпнйяоньйршлкноцщнецуожбуффрщдаештздэжпэряугцебнйжцюебщмльщюдтююндлшежылщзгишньмлудыыфэаецнкллцйигв пттздэсцьвдлцнжбнйфцлезшхьйшпеыхэнзетдяхфплнтрнцмйчэщдцеаеосявьзхьдыхулнгзвепнфеыхыбцэешгнжиьзйшяцсц тбошильэыбжбзблшюшфйэсыкрдщэуэыггвшладзнжддэжллцнеяучщпбчьфбыцшшхымэюшмшдожедддэжосзжкжсявьзоцдлгщьш йшыццэкедэяцоогьцдыысухвдэщджоэозбжшзвдэщдыцтбеднбдьжргцьцжебящцйшюеэнгриршхвэявррэнпзбвешцйэзцецэиш оебябшдицнвхлцжцьиьэйшсныхюрвофбщнсцрндбяэнцхумудлжцуюжцдыхуеиштюкшэмббрьшжшжцбшмшкйрцозыдэянцчьщцйш юеыылшыйяцннильэявррефшлбшожупнцйбйбтйлньцашелдбрццэхьгцвлэтлщубжеисоояцтфжцнцеевльэюпзбюбюшьщмлкбье сдмнялшеддцэюшбойшцшнбяэйуоицщвезхьфджлэсаешлштчщйвшлчфшлжцяужэгурхгцазыпщнишбшвуцбинцымлвэлнщнтйюб мпмбщибищэямжшыэаншштрхутылщузйуямябнцйбнцэщйвддтбозаюодиютбрввшышыцтйкшйшцштрщхйбкбйбдбрцозбузбашыв зцзбоишэйвддпшепнццэефхьвддэгщтылиезхьенпшишгебяяюуебяццэщмэхвмлмдрцозлбрцозмдчэвэцбцлцпышоеьзяяцэкн ыоошцэкещдшэюпбшдицнддцеяууюювмлшсжбгнйрщдйдэтюкйвнцкдбщрхооыицщцсцащуыхвыхшдыцюцоцылрцозыдэянцрцвдвэ аеддьзгтгвмэефоцкшйртщыыыбюебяяцооыцжбньожбвфшувсшыдеишэтцзбщцфедэюпвоххбщдбпшюзебжпэрюриртдбтркеюрщд мнсцлеьшишсщаеезацжедлгноищхюшбфжцгьнцйбтщбишэйвмэйуишыйшщыкзяьзсдкеьнщцплзщбпщнюшшшцняцгеисжоейыдйб ошфбяонрддтбщциепттзцаыдыбтйтруялшхйщцщзгтдлсодэюпюбюроццдьючбшлжшонаддлсучщщоуюлшдыосжбньжбдэчьубхй эзжкьтгтюжтбгньцхйщцхищлгнилтдсцащьтзбишзссдыцтйпцчзяуьттздэтбкштйьнубхйжсанрвяшяцлцнбшлжцхьтщвлйоош оегулшппэцшшхйыцтйоцяолшахлцябнйкевлзолныиюрыбфекюиэефоцкнперддещэяштызбинлррдэюуоизяугианябшэюпщцжб гебящньщштшэщхйбтысцащьтддыылнщедэжцюештщхцечвдэгтгтюжтбтщюфпфжцжцсцкдфыкеэяибтйзцзбтщцедлдэчэефжцжц лряузбйотбозщхштявячыхэныбецтйдэуомнцэьштщыыыбюеьшоигцжщпфхьдлябтбхьшлябтбхьчэццчоибжцрехуюввэщцывар лщцэнццеисощбувефбтйюбщнзлявйшюеисоошшентщжырещхшбзхщлжцплщцяцсцоореушкдмлтьпебяцэхрщцжшцэбвюкцнууббщ ыквегуаэюрмэшэюпсцэцышхьаоинебгжбщдспшюшэщшлжцяцдшгшэдгщтбаэбщмлыькбдэюпфьфцщцнеьтянкшжшцэхщюыкеоцкэ ыбфьйцрэкеяугйнчяятысцорвлхвцнаэьзкношилщиявсхйусцаоыбфйцеяуэглщцэроплкмжцплыдьзоцялябтбйшйбйбошфбяо нрирщхйбхищлгнилтдвнэеуясцащьтюбяюйшэеддгщоцяолшилжцьэфэйугньцхйщцфцплбуьззбпээяоцчшннйшлншлвлнцхуфц ьщщдвехьялвлнцяяуопцжрирсзэзюжтбчилизнщпщпцшеэсзжоындэыкдллцнештдлэзбпцэвеьюйбнегокшцжьсреяяхифрлнтб шнцэгитьчзыклшялюбкбьцюикшлнюеиссцжшенмбюшхьшрхршщуыхьцыгщвешлишоштржжвеоцяцпшзбйбошфбяонргттздэсцпэ бщьэфлоцжбшлоцжббшошболшфеяшецлцппюндльллцжбаэыыроцбиляяцбтрщхйбжддэврдбуозиьзтщдлтриряоуогныишдджцмп нйтбозаювнаопльтбвяшвндьнцплыдюшчшзбфещдзнфещдкцлняоуоилжшябнйохрттбэотьщририрйшозкехьклявйшюебяюбнф фрнцнйчшкноцтыьзсдкеьнщцздцеэнецуожбуффрщдкноцтыьзсдкеьнщцнцлнтщьсьзхвцедлтьпегоьцннинсцаэптдлтоэевэ ыбжбтщрхптйвмлпныбхимпывчвмлмдоцфйаешлщцтйвниннетысцьцкуйужцюевэвеццьвирщдчэштюкйвхвдэлнбуйвчфэжцнуо ювмлвэцэйвтыгтоонннлфыезисупыйвэгьцэгцяцццшчбаевлябоцкннлшеирыбзцощдхкемлйуэяьщяуфшзбсигцкбцэыюцциэтз кекеыхьздссолцьцлцрцэеддвлзщрдюжшэефоцяввецидсщсхэфбпнлябжоеирщнгцкбфевлнйяцгцсщмлкбнлфылшыйзбкдги явщдцеаылшыйщцуфхвмлчгрдхвтыпгапаздлокрддэинцйплфобжупашывушошцбфевлщдгщибдэюпщнквццрлчшеэяячщыклшлн еназдлуигцжцюшоеяшхвсшыйштщвэбщвехьдэгцшьадгщщплфжцгьцэмэсуукьнзбщнхбтйьннрсзявуежлзщкецэнцжбнйплых йбплгктыхйцэыбрладщдмэнзсхдбфццшгецвюпчэмэгцкбщхгцэеэяшьтйоцдычшяятрявоышэшвэзшеяутзяусхлшйыяввэяшжш жесшречимпшшилфыхугбябцшрндрыбтрйввехьюидлцбхьхтгщыхйуяютйтрщхацашыбддаеябфзгщыдыпцхлшхбтйьннрчимпбо илчуюкинцэншпэефоцбшдицнэшышйшжщбвяшрвьзязфрэястлцлншэйуыбцэщэтыюшйэддвэчэирмшелфрщдяущлезхркноцащез лнэощцяцгцрегуащузйущззэюшывйэзбфцлезшлщяомьсцлцнбньдллцнеюржжцнгнащщдэяхбцэямябзщбижылщблибцрлианее уптыкнинцэуппйшилцрнрявьбябтщчууптбылзщбвирчофтвэлнаозщвелнтбцхлшсцтбмьжбйэщцжтэятрсшжнтйдьнцомябиз явущэфыпфзфрэяжшфеэяцбышуоювмлюбтушщхвюпыэвлтбошахлшщцуфешилтхбщьзцепфцеешилчвдлзбтхооцйыэвлтбошахбщ ьзощмлябдэнеэшхвццзбмшюшьцннадяцгцнназддзбцеддаршлючдладтшжеарупйшцвщдхуупащкемхлшыцлюилузишснтьццре кеддюшжпсцщшозешгнюбмвезыдешзигцпшщцзвжчцеэяоомбжрэялшюшнцьршлжцхтдльщзяйшбубщдэыбкнжддэщдрвичдэюпйб

жшщцоиявэснцьщсхьзцеюрьсдшвуйвдэяуашжшхьйианыцспгццэоцсцтлйокбдэылйуяусхбщьзчеефоыкеефьюцбюноочбйбтй
лнгигццэбщжлбштбйэсцщдыбвопльтиношюбщвдлньжбйэяцгцмылиосяунцчьшэыбкняцгцтллцнебищлбшдэчшеэишдэрдоцфй
ысьзшэйыбууюфржкрцеозщацощдхичыхоочкыдащаыешилтдяцгцннащыышэсцюввэшэмхлщжкшэефоцчьмпфьбвьзюцмрунсцфщ
мэнзбивэбихьбпэяуштхвцццэдджшоеяшзбщнлццэнзыкщпчщжыэзешдьгщмбтфцеэятянотрнцмйябиедцощдхзняцгцйбрдюш
чшлнвфйвшлнзуцтсоаыфлщинропуветяуоювмлчшпввеоцкдцецвюпсцзщбицефшщцйшюеьшувьсефижуивлэцзщдэтеезишнбдь
ыхдбнеяшишюнтбхкрддэшшлцишцнтбхкрддэтщчщьзкеезбианкэшлвзирщдыбвоишчйяцгцкдщцугараемцздпшяцинцйыцтйпц
йивэюшкбоебяяомпщнщцыбьвтекебусцлеаелщьзфнплжслцсцбоятюыщлябинфьшшдиюдтюадьзчцщрйвюдтюзщмбтфцешибщэя
тбтбшептгзшжуюпросптлщтблшгнышщцыбьвьлдьжцведлтьпеэяяоюшвдюшчшлнжцйшадэцрфйврдиснцлеосбшвушэщдхушщдд
ншздыбхиоссожцицощжкншйимлгнаэштдцдогннйтбшубврртюдруыйшьнцнэжшщибадмббрьсзщддосжцюевлысупяохбчбаэшт
дцювдссонйсцсибввеэяжцуфзякшйигвунзбщцйшнщыссонйащезяутбэеяячивэюряуууяввриржоибильэжшнчсшсхинщццэыб
юбпквеоцяцащяуяушлцеяяэцлесшиейурхыылщрдкшкэбщошцэнцжшфеислшэщяяябудвэеепфшэаэвэшецввлвзезсхдбнбябие
упнциепфбвщдармлтщчщьзэятыяцйбзхьзшшлцишьидсэчешлфжцжцшиангцчьлиангцйбнещдлтгщдэодьшцефрщхйбяцгцсиае
млжцячбутзюраеебйшсцщшозешадзфоцбусхгцвнятюыэцедзбшцкбфевлнйаенцбщукхвюпщцснйрэяшэщищцицащяухьщноцуе
ыгйуццпиьзюшюбдээоцэбнбикьтррдешппрццдюшреисяотбкшцнвопшлнмцсотщчубитьишадмшюшсиюрешаоишинлгрдхвтыпг
апоцтбозбярцяэщхэнювузтыосжшцэсовыбввеяшнцлебншсбщаеьэзбишцнирщхйблняочйувыгжыбуиншлшшеедэсцувэзлннф
бвщдарсшяцщцснйруннбрцфьбвбимэбнашлшоеяшякддьноцйщмлыьрвйэюпоиыбфхбщьзцбщбрцхбщдыбщзсхдбфцюбйбшшоеяу
рхешжеюрирупошодфысуюкшшлцишщццнтщчщьзэятбьиаеьэчфжцеепфмэаесзжощзфпсщсхирщхйбаебищлгнехдбюбмшэдбяжш
жеуяштмпсиыыезццуожбуффршлштыбхьтыдхвэщджшхьоигцьибщлианцэшимэррщджцгцаооцуоювмлюбцэгвцеэзкношильлчш
цэезсхсцовсзсхяцлцнбюбгнщврдйбжшфшфкрдэятржыйвуоювмлоцаэвэявэоаесшышыцтйкшрнфьзфбуцрирошчэюрирошябщц
рлжцчвнцнноцшшцэишювузшемлнццещхцеыдсндбхзкнсонйюпуудлябцеираезгсцчпеджцадзфгщуоювмлнцядыбщбнлузяуэя
тыцноцмудлбшхщьзшэбвчуунэщмэяужцэщцнсдсзбвьсцпбрацодичвеоцвдшэанйшхбчбвнзббшоешлхьдэыаинуоювмлсцхиин
оцвэишехдбюбоцыхаещхщоошжчщэхвчианщцфещдвесшкбжоснхьтымлщцтйвнинсцнцазысртжлзщяягьцэюшдыбуфпюбишеэбв
млоыослцкбинсцлцжешеяшзбишыьзбкдшэанзбщнябщцицпиьзчпеддэгщябщцнуцбнлщщьзшэбвунжцгждхйвцедэбщбиьллбчш
озчехучиьэлоыезяукшжзсхдбснйшодьзхвювжыцэвелнфеисйодштйябюиыыезрдсцооо цюеэякнзщцежсбщжжупгьдбтбеэце
млтбюбдбхбхбдхгтшщщецвуоювмлцэйпинзензддоцщноцжблцжцощбуцрщхйбзбжшжцсцэокбйэябцндэыаинжшэрсшшэьссшжц
юевэгитьгзыкчйаэжжшщмнзивэгцйолщйуукянзепнашывщиявзбмшэдыыйбоеэяснтьрдпшнцьщсхкдткдэюпчщмлбшиэыбжбчи
вляялшхймшюшинжбдэюпэюйбтыооя цнцдьжбдэзщвэрбкьсцтлйокбдэфзяуйутбошилтхбщьзнцадвэшшдийшятбщжцщнсцщцге
биэялшжеезоцрдашжшоцыубщдэыбневццэкедэюплнплрвдлтблнмпшнуоювмлизжоьымлньнэефоцощвлмугщадбямлвбтйгньщ
млхьжрчщонбоинуьйшплхвосзщаещэфльлчшцэлилифькбинлцидаеябцдюрйшбшоешлхьдэыаинвэзгйбошйшштгщсонлсбюбшш
яэьтвещдяшйоившлтбрнсцмбооклябэцошцйньэевцепфешэтеьойбюббшвфжцнеьсбопладтшцйпшепябтттзжцоигщфьыдчэяв
дэвлцнсодтржхбцнчиштюкйвыцснщцюивэыбишжцюещдвнэепюоцейфьчосолещхунжиьзбшрвщхацлшчвзнсцтлябкнеддэжоаэ
ьзкжшэьцлцрццдвэьщжэьшюроцццлнцэдещэцэдещэливэяупэтхрдешцйьсштыбжебиьзьзлплццц лнфцплбутбылвлцврррдбш
оешлхьээыафпафсдцэроижфльтьэиряотэщдмшэдяшхмжцжшжцэщэзижлтвлфряусцжшлннцаожоейдноигцжщпфоцвдьзыдсзэж
ррирадэноцынцэшитюзиьзсояотэацлнмшюшлцщниныйсзкнцоилюфщэмдмшюшннбшепсцжцаоаэщдэзмднцлшплыдсзшэюпйшяб
ьилитржыйвуоювмлчэацшшхйзщтзоцштьллннйгб

Відкритий текст

мальчикизаулыбалисьисжаромвзялисьзаделоонирвализолотистыецветыцветычтонаводняютвесьмирпереплески
ваютсяслужаекнамощеныеулицытихонькостучатсявпрозрачныеокнапогребовнезнаютугомонуиудержуивсевокр
угзаливаютслепящимсверканиемрасплавленногосолнцакаждоелетоониточносцеписрываютсясказалдедушкапус
тьихянепротиввонихсколькостоятгордыекакльвыпосмотришьнанихподольшетакипрожгутутебявглазахдыркувед
ьпростойцветокможносказатьсорнаятраваниктоеенезамечаетамыуважаемсчитаемодуванчикблагородноерастен
иеонинабралиполныемешкиодуванчиковиунесливнизвпогребвывалилиихизмешковивотьмепогребаразлилосьси
яниевинныйпрессдожидалсяихоткрытыйхолодныйзолотистыйпотоксогрелегодедушкапередвинулпрессповерну
лручкузавертелбыстрейбыстрейипрессмягкостиснулдобычунувотвоттаксперватонкойструйкойпотомвсещедрее
обильнеепобежалпожелобувглиняныекувшинысокпрекрасногожаркогомесяцаемудалипербродитьснялипенуир
азлиливчистыебутылкииздподкетчупаионивыстроилисьрядаминаполкахпоблескиваявсумракепогребавиноизодув
анчиковсамыеэтисловаточнолетонаязыкевиноизодуванчиковпойманноеизакупоренноевбутылкилетоитеперькког
дадугласзналпонастоящемузналчтоонживойчтоонзатемиходитпоземлечтобывидетьиощущатьмиронпонялещеод
нонадочастицувсегочтоонузналчастицуэтогоособенногоднядня сбораодуванчиковтожезакупоритьисохранитьап
отомнастанеттакойзимнийянварскийденькогдавалитгустойснегисолнцаужедавнымдавнониктоневиделиможетб
ытьэточудопозабылосьихорошобыегосновавспомнитьвоттогдаонегооткупоритведьэтолетонепременнобудетлет
омнежданныхчудесинадовсеихсберечьигдетоотложитьдлясебячтобыпослевлюбойчаскогдавздумаешьпробратьс
янацыпочкахвовлажныйсумракипротянутьрукуитамрядзарядомбудутстоятьбутылкисвиномизодуванчиковонобу
детмягкомерцатьточнораскрывающиесяназарецветыасквозьтонкийслойпылибудетпоблескиватьсолнценынешн
егоиюнявзглянисквозьэтовинонахолодныйзимнийденьиснеграстаетизподнегопокажетсятраванадеревьяхоживу
тптицылистваицветысловномириадыбабочекзатрепещутнаветруидажехолодноесероенебостанетголубымвозьми
летоврукуналейлетовбокалвсамыйкрохотныйконечноизкакоготолькоисделаешьединственныйтерпкийглотокпод
несиегокгубамипожиламтвоимвместолютойзимыпобежитжаркоелетотеперьдождевойводыконечноздесьгодится
толькочистейшаяводадальнихозерсладостныеросыбархатныхлуговчтовозносятсяназарекраспахнувшимсянавст
речунебесамтамвпрохладныхвысяхонисобиралисьчистоомытымигроздьямиветермчалихзасотнимильзаряжаяпо
путиэлектрическимизарядамиэтаводавобралавкаждуюсвоюкаплюещебольшенебескогдападаладождемназемлю
онавпиталавсебявосточныйветеризападныйисеверныйиюжныйиобратиласьвдождьадождьвэтотчассвященноде
йствияужестановитсятерпкимвиномдугласхватилковшвыбежалводворииглубокопогрузилеговбочонoксдождево
йводойвотонаводабылаточношелкпрозрачныйголубоватыйшелкеслиеевыпитьонакоснетсягубгорласердцамягко
какласканоковшиполноеведронадоотнестивпогребчтобыводапропиталатамвесьурожайодуванчиковструямиреч

екигорныхручьевдажебабушкавкакойнибудьфевральскийденькогдабеснуетсязаокномвьюгаислепитвесьмириул
юдейзахватываетдыханьедажебабушкатихонькоспуститсявпогребнаверхувбольшомдомебудеткашельчиханьехр
иплыеголосаистонылростуженнымдетямоченьбольнобудетглотатьаносыунихпокраснеютточновишнивынутыеиз
наливкивсюдувдомепритаитсяковарныймикробитогдаизпогребавозникнетточнобогинялетабабушкапрячатото
подвязанойшальюонапринесетэточтотовкомнатукаждогоболящегоиразольетдушистоепрозрачноевпрозрачныес
таканыистаканыэтиосушатоднимглоткомлекарствоиныхвременбальзамизсолнечныхлучейипраздногоавгустовск
огополудняедваслышныйстукколестележкисмороженымчтокатитсяпомощенымулицамшорохсеребристогофейер
веркачторассыпаетсявысоковнебеишелестсрезаннойтравыфонтаномбьющейизподкосилкичтодвижетсяполугам
помуравьиномуцарствувсеэтовсеводномстаканедадажебабушкакогдаспуститсявзимнийпогребзаиюнемнаверноб
удетстоятьтамтихонькосовсемоднавтайномединениисосвоимсокровеннымсосвоейдушойкакидедушкаипапаидяд
ябертидругиетожесловнобеседуястеньюдавноушедшихднейспикникамистеплымдождемсзапахомпшеничныхпол
ейижареныхкукурузныхзеренисвежескошенногосенадажебабушкабудетповторятьсноваисноватежечудесныезо
лотящиесясловачтозвучатсейчаскогдацветыкладутподпресскакбудутихповторятькаждуюзимувсебелыезимывов
севременасноваисноваонибудутслетатьсгубкакулыбкакакнежданныйсолнечный

Програмний код:

```
import string
import re
import math
from operator import itemgetter

alphabet = open("Alphabet.txt",encoding='utf-8')
alphabetData = alphabet.read()
textGeneral = open("Textcp3.txt", encoding='utf-8')
textGeneralData = textGeneral.read()
textGeneralDataFiltred = re.sub(r'[^а-яА-Я ]+', '', textGeneralData)
textGeneralDataFiltred = re.sub(r'\s+', ' ', textGeneralDataFiltred)
text = textGeneralDataFiltred.replace(' ','')
bigrams = ['ст','но','то','на','ен']

def entropyMono(Dict):
    entropyData = 0
    for elem in Dict:
        if Dict[elem] != 0:
            entropyData -= Dict[elem]*math.log(Dict[elem],2)
    return entropyData

def monogramCount(text, alpha):
    monogramDict = monogramDictCreate(alpha)
    for elem in text:
        monogramDict[elem]+=1
    for elem in monogramDict:
        monogramDict[elem]/=len(text)
    return monogramDict

def monogramDictCreate(alpha):
    return {item: 0 for item in alpha}

def Decrypt(text, key, bidict):
    text2 = ''
    for position in range(0,len(text)-1,2):
        a = Equation(key[0], 961, 1)
        if type(a) is int:
            Y = bigramData(text[position]+text[position+1])
            X = (a*(Y-key[1]))%961
            text2 += bidict[X]
    if text2 != '':
        entropy = entropyMono(monogramCount(text2, alphabetData))
        if entropy < 4.5:
            print(key)
            print(text2)


def alfaDictCreate(alpha):
    alfaDict = {}
    for i in range(len(alpha)):
        alfaDict[alpha[i]] = i
    return alfaDict

def Evclid(a, b):  #поиск gsd
```

```
    if b > a:
        a, b = b, a
    while 1:
        q = a // b
        r = a % b
        a, b = b, r
        if r == 0:
            break
    return b

def Inverse(a, m):  #поиск обратного
    if Evclid(a,m) == 1:
        r0, r1 , u0, u1, y0, y1 =a, m, 1, 0, 0, 1
        while r1:
            q = r0 // r1
            r0, r1= r1, r0 % r1
            u0, u1 = u1, u0 - u1*q
            y0, y1 = y1, y0 - y1*q
            if a*u0+m*y0 == 1:
                return u0
        else:
            return 'error'

def Equation(a, m, b):    #решение уравнения a=bx mod m
    r0, r1 , u0, u1, y0, y1 =a, m, 1, 0, 0, 1
    while r1:
        q = r0 // r1
        r0, r1= r1, r0 % r1
        u0, u1 = u1, u0 - u1*q
        y0, y1 = y1, y0 - y1*q
        if a*u0+m*y0 == 1:
            if u0 < 0:
                return ((u0 + m) * b) % m
            else:
                return (u0 * b) % m
    if b % r0 !=0:
        return 'error'
    else:
        u0 = Equation(a//r0, m//r0, b//r0)
        listO = []
        while 1:
            listO.append(u0)
            u0 += m//r0
            if u0 > m:
                return listO

def bigramDictCreate(alpha):
    return {item1+item2: 0 for item1 in alpha for item2 in alpha}

def bigramCount(text, alpha):
    bigramDict = bigramDictCreate(alpha)
    for position in range(0,len(text)-1,2):
        bigramDict[text[position]+text[position+1]] += 1
    for elem in bigramDict:
        bigramDict[elem]/= len(text)//2
    return bigramDict

def writeBigram(text, alpha, n):
    bigramDict = bigramCount(text, alpha)
    bigramList = []
    count = 0
    for i in sorted(bigramDict.items(),key=itemgetter(1), reverse = True):
        count += 1
        bigramList.append(i[0])
        if count == n:
            break
    return bigramList

def bigramData(bigram):
    X = alpha[bigram[0]]*31+alpha[bigram[1]]
    return X

def bigramValueDict(alpha):
    bigramValueDict = bigramDictCreate(alpha)
```

```
    for key in bigramValueDict.keys():
        X = bigramData(key)
        bigramValueDict[key] = X
    bigramValueDict2 = dict((v,k) for k, v in bigramValueDict.items())
    return bigramValueDict2
def foundKey(bigramList, bigrams):
    keys=[]
    for n in range(len(bigramList)):
        for m in range(len(bigramList)):
            for i in range(len(bigramList)):
                for j in range(len(bigramList)):
                    if (n!=m) and (i!=j):
                        Y1 = bigramData(bigramList[n])
                        Y2 = bigramData(bigramList[m])
                        X1 = bigramData(bigrams[i])
                        X2 = bigramData(bigrams[j])
                        a = Equation(X1 - X2, 961, Y1 - Y2)
                        if type(a) is list:
                            for k in a:
                                b=(Y1 - k*X1)%961
                                if [k,b] in keys:
                                    continue
                                keys.append([k, b])
                        else:
                            if a != 'error':
                                b=(Y1 - a*X1)%961
                                if [a,b] in keys:
                                    continue
                                keys.append([a, b])
    return keys

alpha = alfaDictCreate(alphabetData)
bidict = bigramValueDict(alpha)
a = writeBigram(text, alpha, 5)
keys = foundKey(a, bigrams)
for i in keys:
    Decrypt(text, i, bidict)
```

Висновки:

Під час виконання роботи, набули навичок частотного аналізу на прикладі розкриття моноалфавітної підстановки; опанували прийоми роботи в модулярній арифметиці.