ინფომაციის უსაფრთხოება

შესავალი

თანამედროვე მიღწევებმა კომპიუტერულ ტექნოლოგიებში წამოჭრა რიგი პრობლემები, რომლებმაც მოიცვა ადამიანთა მიღწევების თითქმის ყველა სფერო. ამ პრობლემების ძირითად ნაწილს შეადგენს კომპიუტერული უსაფრთხოების უზრუნველყოფა, რომელიც მეცნიერების (სპეციალისტების) კვლევის აქტუალური საგანია და ამასთან ერთად მილიონობით გამოთვლითი სისტემების მომხმარებლების მოთხოვნაცაა.

აღნიშნული პრობლემიდან განსაკუთრებულ ყურადღებას იმსახურებს ორი ამოცანა:

1. გამოთვლითი სისტემების დაცვა არასანქციონირებული შეღწევისაგან,
2. ინფორმაციის (მონაცემების) დაცვა, რომელიც ფუნქციონირებს გამოთვლით სისტემებში.

დღევანდელი შეფასებით „ინფორმაცია“ - ეს არის ძვირადღირებული საქონელი, რომელიც შეიძლება იქნეს შეძენილი, გაყიდული, გაცვლილი და ა.შ. მისი ღირებულება ათჯერ და უფრო მეტად ძვირია იმ სისტემებზე, რომლებშიც ის ფუნქციონირებს. აქედან გამომდინარე, ინფორმაციის დაცვა იყო, არის და რჩება აქტუალურ ამოცანად, რომლის განხილვასაც ეძღვნება წინამდებარე დამხმარე სახელმძღვანელო.

მეცნიერებაში ინფორმაციის დაცვის მიმართულება პრაქტიკაში დამკვიდრდა „კრიპტოგრაფიის“ სახელწოდებით. იგი ცნობილია გასული საუკუნეებიდან და იმდროინდელი მკვლევართა განსაკუთრებულ ყურადღებას იმსახურებდა, რომელთა მიღწევები დღესაც შეიძლება ეფექტურად იქნეს გამოყენებული.

კრიპტოგრაფია ინფორმაციის დასაიდუმლოების სამეცნიერო-ტექნიკური დარგია, რომელსაც განვითარების მრავალსაუკუნოვანი ისტორია აქვს. მან განვითარების განსაკუთრებულ საფეხურს გასული საუკუნის მეორე ნახევარში მიაღწია, როდესაც მისი მეთოდები მათემატიკურ სისტემებს დაეფუძნა, ხოლო 1976 წლიდან ღია გასაღებების მეთოდოლოგიამ მას თვისობრივად ახალი ხარისხი შესძინა. კრიპტოგრაფიის გამოყენების სფერო მრავალმხრივია. მაგალითად როგორიცაა სამხედრო-სახელმწიფოებრივი და საბანკო-საფინანსო კომერციული საქმიანობა, ლოკალური და გლობალური ქსელებში (ინტერნეტში) ფუნქციონირებადი პროგრამებისა და მონაცემების დაცვა და სხვ.

1949 წლამდე კრიპტოგრაფიულ „კვლევებს“ და ამ მიმართულებით ჩატარებულ სამუშაოებს, აგრეთვე მიღებულ შედეგებს განიხილავდნენ როგორც ხელოვნების ნიმუშებად და არა როგორც მეცნიერულ მიღწევებად. ცნობილია იულიუს ცეზარის და ცეზარ ავგუსტის ტექსტის დაშიფვრის მეთოდები, რომლებსაც ისინი იყენებდნენ, ჯერ კიდევ 21 საუკუნის წინ, მეგობრებისადმი გაგზავნილ შეტყობინებებში. ასე მაგალითად, იულიუს ცეზარი (ცეზარ ავგუსტი) საწყისი ტექსტის დასაშიფრად იყენებდა ლათინურ ალფავიტს და მეთოდს, რომლის მიხედვითაც ხდებოდა დასაშიფრი ტექსტის ყოველი სიმბოლოს წანაცვლება სამი (ოთხი) პოზიციით მარჯვნივ ან მარცხნივ და მისი ჩანაცვლება სიმბოლოთი, რომელიც აღმოჩნდებოდა ალფავიტში წანაცვლების შედეგად განსაზღვრულ პოზიციაში.

ყურადღებას იმსახურებს 1926 წელს ამერიკის ტელეფონებისა და ტელეგრაფების კომპანიას ინჟინერის გ.ს. ვერნამის მეთოდი, რომელშიც გამოყენებული იყო ცეზარის მიერ შემოთავაზებული მეთოდი, იმ განსხვავებით, რომ ნაცვლად ლათინური ალფავიტისა, შიფროტექსტის მისაღებად ის იყენებდა ორობით ე.წ. „ბოდო“ კოდს.

ახალი ერა კრიპტოგრაფიაში დაიწყო 1949 წლიდან კ.შენონის ნაშრომის “საიდუმლო სისტემებში კავშირის თეორია” გამოქვეყნებიდან, რომლის საფუძველსაც შეადგენდა მის მიერ 1948 წელს გამოცემული სტატია, რაც თავის მხრივ ინფორმაციის თეორიის განვითარების დასაწყისიც იყო. მიუხედავად კ.შენონის ასეთი დიდი წვლილისა, ჟ.დიფის და მ.ჰელმანის (პარალელურად გამოცემულმა რ.მარკერის) ნაშრომმა „ახალი მიმართულებები კრიპტოგრაფიაში“ 1976 წელს მეცნიერების და სპეციალისტების ფართო მასების ყურადღება მიიპყრო, რამაც განაპირობა მათი აქტიური ჩართვა კრიპტოგრაფიის პრობლემების კვლევაში.

ტერმინი „კრიპტოგრაფია“ ბერძნული სიტყვაა, რომელიც შედგება ორი ნაწილისაგან და ნიშნავს: CRIPTOS - საიდუმლო და LOGOS - სიტყვა.

კრიპტოგრაფიაში განიხილავენ ორ მიმართულებას:

1. კრიპტოგრაფია;
2. კრიპტოანალიზი.

კრიპტოგრაფიის ამოცანაა გადასაცემი (დამუშავებული) ინფორმაციის მაქსიმალური დაცვა, კერძოდ მისი საიდუმლოებისა და მთლიანობის (ნამდვილობის) უზრუნველყოფა.

კრიპტოანალიტიკოსი, რომლისთვისაც როგორც წესი უცნობია ინფორმაციის დაშიფვრის გასაღები, მაგრამ ხელმისაწვდომია მხოლოდ დაშიფრული ტექსტი (შიფროტექსტი) და შესაძლოა დაშიფვრის ალგორითმი, მოქმედებს კრიპტოგრაფის საწინააღმდეგოდ. კერძოდ, ის ცდილობს „გატეხოს“ კრიპტოგრაფის მიერ გამოყენებული დაცვის სისტემა, გაშიფროს შიფროტექსტი ანუ აღადგინოს მისი საწყისი სახე ან/და გაავრცელოს ჭეშმარიტი შეტყობინების მსგავსი (შებრუნებული) სახის ინფორმაცია.

ინფორმაციის კრიპტოგრაფიული დაცვა შეიძლება ორი გზით: პროგრამული და აპარატურული. აპარატურული მიდგომა ძვირადღირებულია, თუმცა გააჩნია დადებითი მხარეებიც. კერძოდ, მისი წარმადობა საგრძნობლად მაღალია, რეალიზაცია მარტივია, შედარებით დაცულია და ა.შ. ამიტომაც იგი ფართოდ და წარმატებულად გამოიყენება დღემდე პრაქტიკაში. ასე მაგალითად, საკმარისია ავღნიშნოთ ის, რომ აშშ-ის მთავრობის პრინციპული მოთხოვნაა სახელმწიფო ორგანიზაციებმა და კერძო სტრუქტურებმა გამოიყენონ კრიპტოგრაფიული აპარატურა, რომლის ძირითად ნაწილს შეადგენს ნაციონალური უსაფრთხოების სააგენტოს (NSA) მიერ შემოთავაზებული ალგორითმების ბაზაზე შექმნილი მოწყობილობები.

გამოთვლით სისტემებში ინფორმაციის დაცვის მასიური გამოყენების ალგორითმები უნდა აკმაყოფილებდეს შემდეგ ძირითად მოთხოვნილებებს:

- დაშიფრული ტექსტის წაკითხვა უნდა იყოს შესაძლებელი, მხოლოდ გამშიფრავი გასაღების გამოყენებით;
- დაშიფვრის გასაღების უმნიშვნელო ცვლილებამ უნდა გამოიწვიოს შიფროტექსტის მნიშვნელოვანი ცვლილება;
- დასაშიფრი ტექსტის უმნიშვნელო ცვლილებამ უნდა გამოიწვიოს შიფროტექსტის მნიშვნელოვანი ცვლილება იმ შემთხვევაშიც კი, თუ დაშიფვრა ხორციელდება ერთიდაიგივე გასაღებით;
- დაშიფვრის ალგორითმის ძირითადი პრინციპები უნდა იყოს საიდუმლოდ დაცულნი;
- დაშიფვრის პროცესი უნდა ექვემდებარებოდეს კონტროლს;
- დასაშიფრი ტექსტისა და შიფროტექსტის სიგრძე (მათში შემავალი სიმბოლოების რაოდენობა) უნდა იყოს ერთმანეთის ტოლი;
- ნებისმიერი დაშიფვრის გასაღები, გასაღებების სიმრავლიდან უნდა უზრუნველყოფდეს ინფორმაციის საიმედო დაცვას;
- გამოყენებული ალგორითმი უნდა უზრუნველყოფდეს, როგორც პროგრამულ ასევე აპარატურულ რეალიზაციის საშუალებას, ამასთან გასაღების სიგრძის ზრდამ არ უნდა გამოიწვიოს ხარისხობრივი დაშიფვრის საიმედოობის და ალგორითმის გაუარესება;
- დაშიფრული ინფორმაციის დეშიფრაცია არ უნდა ხდებოდეს იგივე გასაღებით;
- დაშიფრული ინფორმაციის დეშიფრაცია არ უნდა სრულდებოდეს უკუგზით;
- ინფორმაციის დაშიფრა-დეშიფრაციის პროცესი უნდა იყოს მარტივი და სწრაფი, ხოლო კრიპტოანალიტიკოსისათვის - გაშიფვრა, რეალურ დროში, შეუძლებელი.

კრიპტოგრაფია იყოფა ორ ძირითად მიმართულებად: სიმეტრიული და ასიმეტრიული სისტემები. სიმეტრიულ სისტემებს მიეკუთვნება ისეთი მეთოდები, რომელთა მიხედვითაც ტექსტური ინფორმაციის დაშიფვრა/გაშიფვრა ხორციელდება ერთი ან რამოდენიმე სიმბოლოს (ე.წ. დამშიფრავი სიმბოლო(ებ)ის ან რომელსაც აგრეთვე უწოდებენ დაშიფვრის დახურულ გასაღებს) გამოყენებით. აღნიშნულიდან გამომდინარეობს, რომ სიმეტრიულ სისტემებში გამოიყენება ერთიდაიგივე გასაღები, ინფორმაციის როგორც დასაშიფრად, ასევე მის გასაშიფრადაც. მიუხედავად იმისა, რომ სიმეტრიული სისტემის მეთოდები არიან ნაკლებად დაცულნი, ისინი დღემდე ეფექტურად გამოიყენებიან პრაქტიკაში, რაც განპირობებულია მათი რეალიზაციის სიმარტივით და სასურველი შედეგებს სწრაფი მიღწევით.

სიმეტრიული სისტემების რიცხვს მიეკუთვნებიან პრაქტიკაში ფართედ ცნობილი ცეზარის, ვიჟინერის, ვერნამის და შებრუნებული მატრიცის მეთოდები, რომლების დეტალურ განხილვასაც ეძღვნება შემოთავაზებული დამხმარე სახელმძღვანელო.

ასიმეტრიულ სისტემებში გაერთიანებულია მეთოდები (რომლებიც ავტორების მომავალი კვლევის საგანი იქნება): დიფი-ჰელმან-მერკლეს, კომუტატიური მატრიცული, რაივეს-შამირ-ეიდელმანის, ელ-გამალის, ბრმად ხელის მოწერის, უდავოდ ხელის მოწერის, ციფრული ხელის მოწერის მეთოდები, რიცხვთა წონა და ა.შ.. ამ მეთოდების მიხედვით დაშიფვრის პროცედურები სრულდება ღია გასაღებით, ხოლო გაშიფვრა კი მისი შებრუნებული ანუ ე.წ. დახურული (საიდუმლო) - გასაღებით. ასიმეტრიული სისტემის მეთოდები არის უფრო დაცული, მაგრამ ხასიათდებიან შესრულების დაბალი სიჩქარით. თუმცა უნდა აღინიშნოს, რომ დიდი მოცულობის ინფორმაციის დასაშიფრად, უპირატესობა ასიმეტრიული სისტემის მეთოდებს ენიჭებათ.

დაშიფრული ინფორმაციის მედეგობა (საიმედოობა) დამოკიდებულია შემდეგ მახასიათე-ბლებზე, როგორიცაა:

1. შიფრაცია-დეშიფრაციის დრო/სიჩქარე;
2. გასაღების გენერაციის (არჩევა/გამოთვლა) დრო/სიჩქარე;
3. დახურული გასაღების ამოცნობის დრო/სიჩქარე;
4. გამოყენებული ფუნქცია;
5. გასაღების სიგრძე (ბიტებში);
6. კრიპტოანალიზის სირთულე;
7. გასაღებების სიმრავლე;
8. გაშიფვრის ალბათობა.

უნდა აღინიშნოს რომ ერთ ან რამოდენიმე მახასიათებლისათვის უპირატესობის მინიჭება, ყოველთვის ხორციელდება სხვა დანარჩენი მახასიათებლების ხარჯზე.

დასასრულს შევნიშნოთ, რომ წინმდებარე დამხმარე სახელმძღვანელოს ძირითადი მიზანია დაეხმაროს ბაკალავრიატის სტუდენტებს, მაგისტრანტებს, დამწყებ სპეციალისტებს სიმეტრიული მეთოდების შესწავლით შეიძინონ საკმაოდ ღრმა ცოდნა, რაც მისცემს მათ საშუალებას დასვან და გადაჭრან კრიპტოგრაფიისათვის დამახასიათებელი თანამედროვე ამოცანები და პრობლემები. შეიმუშავონ ახალი მეთოდები და მათი რეალიზაციის საშუალებები, პრინციპები და გზები. ხოლო ინფორმაციის დაშიფვრა/გაშიფვრის შემოთავაზებული პროგრამული მოდულების გამოყენება დაეხმარება მათ სხვადასხვა სახის პრაქტიკული ექსპერიმენტების ჩატარებაში.

თავი 1
ინფორმაცია, ენტროპია და განუსაზღვრელობა

ინფორმაცია ცხოვრების აუცილებელი ატრიბუტია. ადამიანი ცხოვრობს ინფორმაციულ გარემოში და გამუდმებით მონაწილეობს ინფორმაციულ პროცესებში.

ინფორმაციული ეწოდება ისეთ პროცესს, რომელიც დაკავშირებულია ინფორმაციის მიღებასთან, შენახვასთან, გარდაქმნა-ანალიზსა და გადაცემასთან. ასეთ პროცესს ინფორმაციის დამუშავების პროცესსაც უწოდებენ.

ინფორმაციის მოცულობისა და ინფორმაციული პროცესების ინტენსივობის ზრდამ ინფორმაციის ავტომატური დამუშავების აუცილებლობა წარმოშვა. ინფორმაციის დამუშავების უნივერსალურ ინსტრუმენტად კომპიუტერი იქცა.

ტერმინი „ინფორმაცია“ წარმოდგება ლათინური სიტყვიდან information, რაც ნიშნავს ცნობას, შეტყობინებას რაიმეს შესახებ. უფრო დაზუსტებით შეიძლება განვმარტოთ: ინფორმაცია არის შეტყობინებათა ერთობლიობა ობიექტების, მოვლენებისა და პროცესების შესახებ, რომელიც მიღების, შენახვის, გარდაქმნისა და გადაცემის ობიექტია.

ადამიანის ფსიქიკაზე დაკვირვებისას საქმე გვაქვს ინფორმაციის მიღებასა, დამუშავებასა, შენახვასა და გადაცემასთან. ინფორმაცია არის ცნობა სინამდვილის შესახებ, რომელიც მიიღება ან გადაიცემა სიგნალის საშუალებით. სიგნალი კი არის ფიზიკური ფაქტი, რომელსაც აღნიშნული ფუნქცია ახასიათებს (მაგ.: ასოები, ნახატი, სიტყვა, სინათლის ტალღები და სხვ.).

ინფორმაცია არ არის მატერიალური ან ენერგეტიკული ცნება. მისი არსება იმაშია, რომ მიმღებს უბიძგებს განსაზღვრული ქცევის არჩევაში _ განსაკუთრებით აზროვნების სფეროში. ადამიანი ინფორმაციას ღებულობს სამი არხის საშუალებით:

ა. მემკვიდრეობითი ფაქტორებით (გენებით), რომლითაც ადამიანს მშობლებისაგან გადაეცემა ნივთიერი და სტრუქტურული ნიშნებით;

ბ. ადამიანებისაგან გადაეცემა ჩამოყალიბებული აზრებისა და მითითებების სახით (მაგ.: წიგნის საშუალებით);

გ. უშუალოდ გარემო სინამდვილისაგან.

ცხოველებისაგან განსხვავებით, რომლისთვისაც მემკვიდრეობითი ინფორმაცია განმსაზღვრელია, ადამიანისათვის მეტად დიდი მნიშვნელობა აქვს მეორე და მესამე არხით მიღებულ ინფორმაციებს. მათ ადამიანი ღებულობს სწავლისა და შრომის საშუალებით, ურომლისოდაც იგი ვერ ჩამოყალიბდება პიროვნებად.

აუცილებელ ინფორმაციათა მარაგით ადამიანის აღჭურვის თვალსაზრისით განსაკუთრებული მნიშვნელობა აქვს სკოლას, რომელიც მიზნობრივად ორგანიზებულ გარემოს წარმოადგენს. ყოველ სასწავლო საგანში მოსწავლე ითვისებს, ინახავს და შემდეგ თავის საქმიანობაში იყენებს განსაზღვრულ, ამ საგნისათვის დამახასიათებელ ინფორმაციებს.

ინფორმაცია სხვადასხვა მოვლენისა და ობიექტის შესახებ განსხვავებული შინაარსისაა. მეცნიერული მიდგომა ინფორმაციის ცნებისადმი მის შინაარსზე არ არის დამოკიდებული და საშუალებას გვაძლევს რაოდენობრივად შევაფასოთ იგი. ასეთი მიდგომა ინფორმაციას განიხილავს, როგორც რაიმეს ცოდნის განუსაზღვრელობის საზომს ანუ არცოდნის შემცირების საზომს.

ვთქვათ, ვაგდებთ მონეტას სწორ ზედაპირზე. ამ დროს, ორი შესაძლო მოვლენიდან, ერთ–ერთი მონეტა დაეცემა პირით ან ზურგით. მონეტის აგდების წინ არსებობს განუსაზღვრელობა იმის შესახებ, თუ რომელი ზედაპირით დაეცემა მონეტა. მას შემდეგ, რაც მონეტა დაეცემა ანუ მოვლენა განხორციელდება, დგება სრული განსაზღვრულობა – მივიღებთ ინფორმაციას მისი შედეგის შესახებ. ამ შემთხვევაში შეიძლება ითქვას, რომ მიღებულმა ინფორმაციამ თავდაპირველი განუსაზღვრელობა ორჯერ შეამცირა, რადგან ორი შესაძლო შედეგიდან ერთი მივიღეთ. შეიძლება განვიხილოთ მოვლენები, რომელთაც ორზე მეტი შესაძლო შედეგი აქვს. მაგალიტად, კამათლის აგდების შემდეგ მივიღებთ შეტყობინებას ექვსი შესაძლო შედეგიდან ერთ–ერთზე. რაც მეტია მოვლენაში შესაძლო რაოდენობა, მით მეტია თავდაპირველი განუსაზღვრელობა და უფრო მეტი რაოდენობის ინფორმაციას შეიცავს შეტყობინება ასეთი მოვლენის შესახებ. ვინაიდან ინფორმაციას გააჩნია რაოდენობრივი მხარე, უნდა შეგვეძლოს მისი შეფასება ანუ გაზომვა.

1.1.ინფორმაციის გაზომვა, რაოდენობრივი ზომა

ინფორმაციის საზომი ერთეულია ბიტი. ეს არის ინფორმაციის რაოდენობა, რომელსაც შეიცავს შეტყობინება ორი შესაძლო შედეგის მქონე მოვლენის შესახებ. შეიძლება აგრეთვე ითქვას, რომ ბიტი ინფორმაციის ის რაოდენობაა, რომელიც მოვლენის შედეგის განუსაზღვრელობას ორჯერ ამცირებს.

მონეტის აგდების შედეგად მიღებული შეტყობინება შეიცავს სწორედ ერთ ბიტ ინფორმაციას, რადგან მისი შესაძლო შედეგების რაოდენობა ორის ტოლია.

იმისათვის, რომ შევაფასოთ ინფორმაციის რაოდენობა ორზე მეტი შესაძლო შემთხვევაში, საჭიროა მისი რაოდენობისა და შესაძლო შედეგების რიცხვის დაკავშირება. ეს კავშირი აისახება ფორმულით: $N=2^I$, სადაც I არის უნფორმაციის რაოდენობრივი ზომა, ხოლო N ყველა შესაძლო შემთხვევათა რაოდენობა ანუ სიმრავლე. აქედან გამომდინარე, აღნიშნული ფორმულიდან, ინფორმაციის რაოდენობრივი ზომა განისაზღვრება ლოგარითმით: $I = \log_2 N$. ორი შესაძლო შედეგის მქონე (N-2) მოვლენის ინფორმაციის რაოდენობის შეფასებისას მივიღებთ: $I = \log_2 N = \log_2 2 = 1$ ბიტი, რაც შეესაბამება საზომი ერთეულის ბიტის განმარტებას.

შეტყობინება ოთხი შესაძლო შედეგის მქონე მოვლენის შესახებ შეიცავს ორ ბიტ ინფორმაციას, რადგან $\log_2 4=2$. შეტყობინება კამათლის აგდების შედეგის შესახებ შეიცავს $\log_2 6$ ბიტ ინფორმაციას და ა.შ.

ერთი შესაძლო შედეგის მქონე მოვლენისათვის $I = \log_2 1 = 0$, ე.ი. შეტყობინება ამგვარის მოვლენის შესახებ საერთოდ არ შეიცავს ინფორმაციას, რადგან მისი შედეგი წინასწარაა განსაზღვრული.

ინფორმაციის ლოგარითმული ზომა უნივერსალური საზომია, რადგან ითვალისწინებს მხოლოდ მოვლენის შესაძლო შედეგების რაოდენობას და მის შინაარსზე არ არის დამოკიდებული.

„ინფორმაციის“ ცნება ერთ–ერთი ძირითადია ინფორმაციული სისტემების განხილვისას. არსებობს ინფორმაციის მრავალი განსხვავებული თვალსაზრისი და განსაზღვრებები ზოგადი ფილოსოფიურიდან (როგორც რეალური სამყაროს ასახვა–გამოხატულება) ყველაზე კერძო პრაქტიკულ განსაზღვრებამდე (როგორც–ცნობები, რაც გამიზნულია ინფორმაციის დამუშავების, შენახვისა და გადაცემისათვის). მნიშვნელოვანია ნ.ვინერის აზრი, რომ „ინფორმაცია არის ინფორმაცია და არა მატერია ან ენერგია“ [ვინ].

ამ საკითხების განხილვა–წარმოჩენა სცილდება სასწავლო საგნის მიზნებს და ამიტომ ქვემოთ განხილულია საკითხები ინფორმაციის რაოდენობრივი განსაზღვრებისა (კლასიკური გაგებით), რაც კოდირების ოპტიმალობასა და გადაცემის საიმედოობასთან არის დაკავშირებული.

ინფორმაციის გამოხატვის საშუალებები მატერიალურ–ენერგეტიკულია და ჩვენს შემთხვევაში წარმოადგენს ფიზიკურ სიგნალებს, ასო–ნიშნებს, ციფრულ ჩანაწერებს, ფიგურებსა და სხვ.

საუბარი ინფორმაციის დაცვის სისტემის შესახებ ფორმალურია, თუ ის ინფორმაციის არსებობის თვით ფაქტს არ ითვალისწინებს. ნებისმიერი კოდი, ინფორმაციის მატარებელია. კოდირების სისტემები ინფორმაციის დამუშავების, გადაცემისა და დამახსოვრებისათვის იქმნება, მაგრამ ამით ლოგიკური (გენეტიკური) და სხვა კოდებისაგან განსხვავებით ინფორმაციის დაცვისათვის გამიზნული კოდების მთავარი ფუნქცია მიმდინარე პროცესების დაცვაა გარეშე ზემოქმედებისაგან. მიუხედავად ამისა არ უნდა დაგვავიწყდეს, რომ ინფორმაციის გადაცემისას ბუნებასა თუ ცოცხალ გარემოში, ტექნიკურ სისტემებსა თუ საზოგადოებაში, როგორიც არ უნდა იყოს მისი წყარო თუ მომხმარებელი, იქმნება ინფორმაციული გარემო, რომელშიც კოდირების სისტემები საერთო ინფორმაციულ პროცესს ექვემდებარებიან, ხოლო ინფორმაციული სინერგულობა (ურთიერთშეთანხმებულობა, ურთიერთთანწყობა) თვისობრივად მსგავს და პროცესებისათვის საერთო სტრუქტურათა წარმოქმნას განაპირობებს, რაც განსაკუთრებით კოდირების სისტემებით აისახება და ვლინდება.

1.2.ინფორმაცია და მისი რაოდენობის განსაზღვრება. ინფორმაციის ოპტიმალური კოდირება და გადაცემის საიმედოობა

გარე სამყაროს მთლიანობის ერთ-ერთი ნიშანდობლივი თვისება მისი ინფორმაციული მთლიანობაა.

ინფორმაციის გადაცემისას (ბუნებასა, ტექნიკურ სისტემებსა თუ საზოგადოებაში) იქმნება ინფორმაციული გარემო, რომელშიც კოდირების სისტემები საერთო ინფორმაციულ პროცესს ექვემდებარებიან, ხოლო ინფორმაციული სინერგიულობა (ურთიერთშეთანხმებულობა, ურთიერთთანწყობა) თვისობრივად მსგავს და საერთო სტრუქტურათა წარმოქმნას განაპირობებს, რაც განსაკუთრებით კოდირების სისტემებით აისახება და ვლინდება.

აქედან გამომდინარე, ნებისმიერი კოდი მოიცავს გარკვეულ ინფორმაციას. ზოგი მათგანი სტრუქტურული და ფუნქციონალური თვისებისაა (მაგალითად, გენეტიკური), ზოგი კი გამიზნულია მხოლოდ ინფორმაციის დამუშავებისა და გადაცემისათვის და სხვა, თუმცა მკვეთრი საზღვრის გავლება მათ შორის ხშირად შეუძლებელიცაა.

კოდირების წინამდებარე საგანი ინფორმაციის დაცვისადმია განკუთვნილი, ამიტომ ბუნებრივი და მიზანშეწონილია გარკვეული ფორმით თუ ოდენობით განხილული იყოს ინფორმაციის თეორიის ძირითადი საფუძვლები და შედეგები.

არსებობს "ინფორმაციის" მრავალი განსხვავებული განსაზღვრება ზოგად ფილოსოფიურად (როგორც-რეალური სამყაროს ასახვა-გამოხატულება) ყველაზე კერძო პრაქტიკულ განსაზღვრებამდე (როგორც – მონაცემები, გამიზნული მათი დამუშავების, გადაცემისა და დამახსოვრებისათვის). აღსანიშნავია ტერმინ "ინფორმაციის" მრავალმხრივი (უშუალო, მაგრამ ხშირად გაუცნობიერებელი) გამოყენება ყოველდღიურ ცხოვრებაში: "ინფორმაციული საშუალებები", "მიღებული ინფორმაცია", "არასრული ინფორმაცია", "ინფორმაციული ომი", "ინფორმაციული გარემო" და ა.შ., რაც წარმოადგენს ტერმინ "ინფორმაციასთან" დაკავშირებულ ცნებათა არასრულ ჩამონათვალს. მნიშვნელოვანია ნორბერტ ვინერის აზრი, რომ "ინფორმაცია არის ინფორმაცია და არა მატერია ან ენერგია". ამ საკითხების განხილვა სცილდება სასწავლო საგნის მიზნებს, ამიტომ პირველ თავში განხილულია მხოლოდ საკითხები ინფორმაციის რაოდენობის განსაზღვრებისა (მისი კლასიკური გაგებით), ასევე, – საკითხები ინფორმაციის კოდირების ოპტიმალობისა და გადაცემის საიმედოობის შესახებ).

1.3.ენტროპია და ინფორმაცია. ენტროპია, როგორც განუსაზღვრელობის ხარისხი

ინფორმაციის წარმოქმნის პროცესი დაკავშირებულია რაიმე საგნის (ობიექტის) ან საზოგადოდ სისტემის სინთეზსა და ცვალებადობასთან. შესაძლოა იცვლებოდეს საგნის მდებარეობა ან ფორმა, ადგილი ჰქონდეს ვნობილ ბუნებრივ მოვლენებს (მოღრუბლულობა, წვიმა, ქარი და სხვ.), ყოველივე მას, რაც მოსალოდნელია, მაგრამ დანამდვილებით მათი მოხდენის წინასწარი ხედვა შეუძლებელია, რადგან დაკავშირებულია გარკვეულ შემთხვევით მოვლენებსა და ხდომილობებთან. ესაა ნებისმიერი სფერო

იქნება ის ინფორმაციული საშუალებები, გასართობი თამაშობანი, თუ ჩვეულებრივი საყოფაცხოვრებო გარემოებები.

შესაძლოა, ჩვენ თავად ვატარებდეთ ცდას, დაკავშირებულს შემთხვევით ხდომილობებთან (სიდიდეებთან). ყოველ ასეთ შემთხვევაში ვარაუდი ამა თუ იმ ხდომილობის მოხდენისა ეყრდნობა ხდომილობათა ალბათობებს. ის რომ, მაგალითად, კამათელის გაგორების შემთხვევაში რომელიმე კონკრეტული x_i-რი, სადაც i = 1, 2, ... , 6, რიცხვის მოსვლის ალბათობა ცნობილია (და ამ შემთხვევაში $p(x_i) = {}^1/_6$-ს), გვაძლევს შესაძლებლობას შევიქმნათ მოცემული ხდომილობის მოხდენის შესახებ. ე.ი. ხდომილობის ალბათობა განსაზღვრავს ჩვენი ვარაუდის ხარისხს სათანადო ხდომილობის შესახებ. შეიძლება დაისვას კითხვა, – როგორია კავშირი ხდომილობის ალბათობასა (ანუ ვარაუდსა) და მიღებულ ინფორმაციას შორის (ხდომილობის მოხდენის შემთხვევაში)? პასუხი კითხვაზე.

განვიხილოთ განსხვავებული შემთხვევა. ის ფაქტი, რომ ხვალ დედამიწა არ შეწყვეტს ბრუნვას თავისი ღერძის გარშემო, დღეისათვის ყველასათვის ცნობილია და ამიტომ ხვალინდელი დღე – დედამიწის შემობრუნება 360^0–ით სრულიად მოსალოდნელია და ამ გაგებით შეიძლება ასევე ითქვას, რომ ახალ ინფორმაციასაც არ შეიცავს (თუმცა არსებობს საწინააღმდეგოს გარკვეული ალბათობა, რაც ყოველდღიურობაში, მისი სიმცირისა გამო, ჩვეულებრივ უგულებელყოფილია).

1.4.ენტროპია და ინფორმაცია. ენტროპია, როგორც განუსაზღვრელობის ხარისხი

შევთანხმდეთ, რომ ინფორმაციის წარმოქმნის პროცესი დაკავშირებულია გარკვეული საგნის (სისტემის, ანუ, საზოგადოდ,– იბიექტის) სინთეზსა და ცვალებადობასთან. შესაძლოა იცვლებოდეს ობიექტის მდებარეობა ან ფორმა, ადგილი ჰქონდეს ბუნებრივ მოვლენებს (ვთქვათ, უღრუბლო ცაზე ღრუბლების გამოჩენა, წვიმის ან ქარის ინტენსიურობის ცვლილება, კამათლის გაგორების შემთხვევაში გარკვეული რიცხვის მოსვლა. ჩვენთვის, ცხადია, საინტერესო მაგალითს წარმოადგენს ინფორმაციის წყაროს გამოსავალზე გარკვეული სიგნალის გამოჩენა და სხ.). ყველა ამ შემთხვევაში ვგულისხმობთ, რომ მოსალოდნელი ცვლილება შეიძლება დასრულდეს ორი ან მეტი შედეგიდან ერთ-ერთით (თუ მდგომარეობის უცვლელად შენარჩუნებასაც ერთ-ერთ შესაძლო შედეგად მივიღებთ), რომლის წინასწარ გამოცნობა შეუძლებელია; შეუძლებელია, თუ საქმე გვაქვს შემთხვევით მოვლენებსა და ხდომილობებთან ისე, როგორც მათ ალბათობის თეორია განიხილავს.

მაშასადამე, საკითხის ასე დასმის დროს არ გვაინტერესებს, თუ რა ინფორმაციას მოიცავს ობიექტი მასზე დაკვირვებამდე. გვაინტერესებს, თუ რა რაოდენობის ინფორმაციას შეიძენს დამკვირვებელი დაკვირვების (ანუ ობიექტის ცვლილების) შედეგად, ე.ი. ინფორმაცია, განსახილველი მიდგომის პირობებით, მიიღება შემთხვევით ხდომილობებზე დაკვირვების დროს.

დავუშვათ, რომ ვატარებთ ცდას, რომლის შედეგი შემთხვევითი ხდომილობებია. დამკვირვებლის ვარაუდი ხდომილობის მოხდენის (ანუ ცდის შედეგის) შესახებ ეყრდნობა ხდომილობათა ალბათობებს. მაგალითად, კამათლის გაგორებისას რომელიმე x_i-ური (i=1, .. 6) რიცხვის გაჩენის ალბათობა ცნობილია და P(x)=1/6-ს. შესაბამისად, – ხდომილობების ალბათობები განსაზღვრავს ცდის შედეგის გაურკვევლობას (მის განუსაზღვრელობას), რასაც ჩვენი ვარაუდი ემყარება. შეიძლება დაისვას კითხვა – რა რაოდენობის ინფორმაციას ღებულობს დამკვირვებელი, როგორც ინფორმაციის მომხმარებელი, ცდის ერთ-ერთი შედეგის შემთხვევაში, ანუ რა რაოდენობის ინფორმაცია მიიღო მომხმარებელმა, როდესაც შემთხვევითი ცვლილება გარკვეული შედეგით დასრულდა? რადგან ცდის განუსაზღვრელობა დამოკიდებულია ხდომილობის ალბათობებზე, მაშასადამე ცდის შედეგად მიღებული ინფორმაციაც დამოკიდებულია ხდომილობის ალბათობებზე და ინფორმაციის რაოდენობა არის ხდომილობების ალბათობათა ფუნქცია).

განვიხილოთ განსხვავებული მაგალითი. დედამიწა რომ არ შეწყვეტს ბრუნვას თავისი ღერძის გარშემო, დღეს საზოგადოდ ცნობილია და ხვალინდელი დღე – დედამიწის შემობრუნება 360^0-ით – სრულიად მოსალოდნელია და, ამიტომ, ახალ ინფორმაციას არ შეიცავს (თუმცა თეორიულად არსებობს საწინააღმდეგო შედეგის ალბათობა, რაც მისი სიმცირის გამო ჩვეულებრივ უგულებელყოფილია). როგორც ვხედავთ, თუ შესაძლო ცვლილების ალბათობა ერთის ტოლია, მაშინ შედეგს ახალი ინფორმაცია არ გააჩნია.

მაშასადამე, ინფორმაციას არ წარმოქმნის ხდომილობა ერთის ტოლი ალბათობით – აუცილებელი ხდომილობა (ანუ ობიექტი, რომლის ცვლილების შედეგი წინასწარ ცნობილია) და, ცხადია, არც ხდომილობა, რომლის ალბათობა ნოლის ტოლია – შეუძლებელი ხდომილობა (ანუ ობიექტი, რომელიც არ იცვლება).

ზემოაღნიშნულის თანახმად, დამკვირვებელმა ინფორმაცია შეიძლება მიიღოს მხოლოდ ცვალებად გარემოზე დაკვირვების შედეგად და არა გარემოში, რომელიც ცვლილებას არ განიცდის ან ცვლილებები დეტერმინირებული ხასიათისაა (დეტერმინირებულია ცვალებადობები, როდესაც ობიექტის ერთი მდგომარეობიდან მეორეში გადასვლა წინასწარ ცნობილ კანონს ემორჩილება).

აღსანიშნავია, რომ ინფორმაციის წარმოქმნა დამოკიდებულია როგორც დასაკვირვებელ მოვლენაზე, აგრეთვე თვით დამკვირვებელზეც. შესაძლოა, ერთი დამკვირვებლისათვის ცვალებადობები შემთხვევითი ხასიათისაა და ინფორმაციული, მაგრამ მეორესთვის – დეტერმინირებული და, მაშასადამე, არაინფორმაციული.

ამრიგად, ინფორმაციის რაოდენობის განსაზღვრის საფუძველი არის ობიექტის შესაძლო ცვლილებათა შედეგის განუსაზღვრელობა, ანუ – არასრული ინფორმაციულობა. ცხადია, ვინც ყველაფერი იცის, ახალ ინფორმაციას ვერ შეიძენს, ე.ი. ინფორმაციის რაოდენობასა და განუსაზღვრელობის რაოდენობას (განუსაზღვრელობის ხარისხს) შორის შეიძლება ტოლობის ნიშანი დაისვას).

რა თვისებებს უნდა აკმაყოფილებდეს განუსაზღვრელობის ფუნქცია? ზემოთ ავღნიშნეთ, რომ განუსაზღვრელობა და ინფორმაციის რაოდენობა დამოკიდებულია ხდომილობათა ალბათობებზე. რაც უფრო მცირეა ხდომილობის ალბათობა, მით მეტია მისი მოხდენის შემთხვევაში მიღებული ინფორმაცია: ზაფხულის დღეებში თოვლის მოსვლას დიდი რაოდენობის ინფორმაცია გააჩნია. ხდომილობის ალბათობის მაღალი მნიშვნელობის დროს შესაბამისი შედეგი ნაკლებად ინფორმაციულია: ზამთრის დღეებში თოვლის მოსვლას ნაკლები ინფორმაცია გააჩნია.

როდესაც მეგობრისაგან ბარათს ხშირად ვღებულობთ, მაშინ მორიგი ბარათის მიღებას (კერძოდ, თვით ფაქტს ბარათის მიღებისას და არა მის შინაარს, – ამ საკითხს ქვემოთ 1.2 განაკვეთში დავუბრუნდებით) არ ახლავს დიდი რაოდენობის ინფორმაცია განსხვავებით იმ შემთხვევისაგან, როდესაც ბარათის ავტორისაგან დროის ხანგრძლივ პერიოდში არავითარი ინფორმაცია არ არსებობდა. ბარათის მიღება ამ შემთხვევაში ნაკლებად მოსალოდნელია და შედარებით დიდი რაოდენობის ინფორმაციას შეიცავს.

ზემოთ განხილულ მაგალითებში ძირითადი ყურადღება ექცევა ხდომილობათა ცალკეულ ალბათობებს, რაც არის საკითხის ერთი მხარე (აქ, კერძოდ, გასათვალისწინებელია შემდეგი გარემოება – რადგან პროცესი შემთხვევითი ხასიათისაა, საჭიროა განისაზღვროს განუსაზღვრელობისა და შესაბამისი ინფორმაციის საშუალო მნიშვნელობა). მართლაც, თუ მხოლოდ ხდომილობათა ცალკეულ ალბათობებს მივიღებთ მხედველობაში და დავუშვებთ, რომ x_i-ური ხდომილობის ალბათობა $P(x_i)\neq 0$, მაშინ მიღებული ინფორმაცია შეიძლება წარმოვიდგინოთ როგორც $P(x_i)$ ალბათობის $I(x_i) = f(P(x_i))$ ფუნქცია და შემდეგი სახით ჩავწეროთ:

$$I(x_i) = 1/ P(x_i) \qquad (1.1.1)$$

(1.1.1) ფუნქციის მიმართ საყურადღებოა შემდეგი არსებითი შენიშვნა. განუსაზღვრელობის ფუნქცია დამოკიდებული უნდა იყოს არა ცალკეულ შედეგთა ალბათობაზე, არამედ ხდომილობათა სრული სისტემის ალბათობებზე, რაც სისტემის ენტროპიის ზომის განსაზღვრისათვის არის აუცილებელი. ქვემოთ ცხადი გახდება, რომ (1.1.1) დამოკიდებულება არასრული და წინააღმდეგობრივია, თუმცა მასში რაციონალურია განუსაღვრელობისა და ხდომილობის ალბათობის ურთიერთდაკავშირებულობა (იგულისხმება, რომ ხდომილობები ადგენენ ხდომილობათა სრულ ჯგუფს და შესრულებულია ნორმირების აქსიომა.)

აღნიშნული საკითხის განხილვამდე განვიხილოთ რაოდენობრივ და სემანტიკურ ინფორმაციათა არსებითი განსხვავებულობა.

1.5. რაოდენობრივი და სემანტიკური ინფორმაცია

წინამდებარე საგნის თემატიკა არ ითვალისწინებს ინფორმაციის სემანტიკური (შინაარსობრივი) მნიშვნელობის განხილვას. სემანტიკური ინფორმაციის საკითხი რთული და პრობლემატურია და არ არის სრულად დამუშავებული (სემანტიკური ინფორმაციის რაოდენობის, როგორც სელექტიური ინფორმაციის განსაზღვრა და მისი სპეციალურ ფუნქციათა სახით წარმოდგენა კერძო შემთხვევებში სავსებით რეალურია).) მაგრამ რაოდენობრივ და სემანტიკურ ინფორმაციათა შორის განსხვავების აღნიშვნა და მისი გარკვევა საკუთრივ რაოდენობრივი ინფორმაციის ფორმულირების გაცნობიერებისათვის არის აუცილებელი.

განსხვავება თითქმის ყოველთვისაა. მას ძირითადად დამკვირვებელი, ანუ მომხმარებელი, განაპირობებს, რაც ჩანს ზემოთმოყვანილი მაგალითებიდანაც. კამათელზე რომელიმე რიცხვის, ვთქვათ, 6-ის გამოჩენა დამკვირვებლისათვის ერთ შემთხვევაში მეტად მნიშვნელოვანი,

საყურადღებო და ინფორმაციულია მაშინ, როდესაც სხვა შემთხვევაში – ნაკლებად მნიშვნელოვანი. იგივე ითქმის მეგობრისაგან მიღებული ბარათის შემთხვევაში: ერთ ბარათს მიმღებისათვის (როგორც ადრესატისათვის), შესაძლოა, განსაკუთრებული მნიშვნელობა ჰქონდეს, სხვა კი იყოს ნაკლებად საგულისხმო და შინაარსიანი, მიუხედავად იმისა, თუ რამდენად ხშირად ღებულობს ბარათს ადრესატი. განსაკუთრებული მნიშვნელობა აქვს მომხმარებლის განწყობასაც განურჩევლად იმისა, თუ რას აღიქვამს; შედეგი შესაძლოა სრულიად განსხვავებული იყოს.

ერთი მკითხველისათვის ან მსმენელისათვის რაიმე წიგნი თუ მუსიკალური ნაწარმოები შინაარსიანი და ემოციური შეიძლება იყოს, მეორესთვის კი – არა. ამიტომ ნებისმიერი მოვლენის თუ ხდომილობის სემანტიკური ინფორმაციის მნიშვნელობა და მისი "ტევადობა" მრავალ ფაქტორზე და განსხვავებულ მომხმარებელზე დამოკიდებული სიდიდეა (სუბსტანტია) და ამდენად ცალსახა ზოგად ობიექტურ გაზომვას არ ექვემდებარება.

ვთქვათ, ერთიდაიგივე კოლოფის, ყუთის, თუ ჩანთის მოცულობის რაოდენობრივი (განზომილებითი) სიდიდის დადგენა შეიძლება, მაგრამ მასში არსებული საგნების სემანტიკური მნიშვნელობის დასადგენად ობიექტური შეფასების მიღება სხვადასხვა მომხმარებლის შემთხვევაში, საზოგადოდ, შეუძლებელია.

ერთი სიტყვით, ერთიდაიმავე მოვლენაში, თუ კოდურ გამოსახულებაში, სხვადასხვა დამკვირვებელმა შეიძლება დაინახოს არა მხოლოდ სხვადასხვა აქტუალობის შემცველი ინფორმაცია, არამედ ინფორმაცია, განსხვავებული თავისი შინაარსითაც, რაც მოვლენის განსხვავებულ შეფასებას (აღქმას) განაპირობებს.

შემდგომში ინფორმაციის განსაზღვრისას განიხილება ინფორმაციის რაოდენობრივი, არასემანტიკური, ასპექტი, რაც დაკავშირებულია ხდომილობათა სრულ სისტემასთან განურჩევლად იმისა, თუ რა შინაარსისა და დანიშნულების ინფორმაციულ პროცესს წარმოადგენს მოცემული სისტემა.

1.6.ენტროპიისა და ინფორმაციის რაოდენობის ზომა

ინფორმაციის რაოდენობის ზომა დაკავშირებულია ობიექტის ენტროპიის (განუსაზღვრელობის) ზომასთან, რაც (როგორც ზემოთ იყო აღნიშნული) ეფუძნება ხდომილობათა სრული სისტემის ალბათობებს.

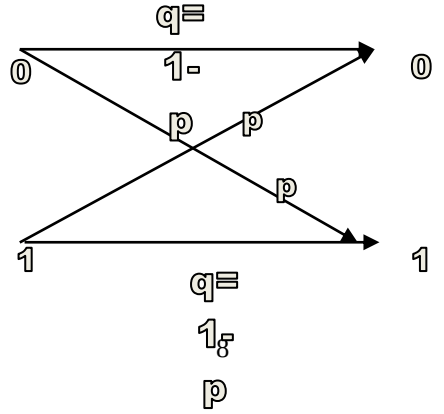
საკუთრივ ინფორმაციის განსაზღვრამდე განვიხილოთ ინფორმაციის გადაცემის პრინციპული სქემა კავშირის დისკრეტული ორობითი სიმეტრიული არხის მეშვეობით (რათა ყურადღება გამახვილდეს ინფორმაციის გადაცემის პროცესის ფიზიკურ მხარეზე). დავუშვათ, რომ დროის გარკვეულ t მონაკვეთში გადაიცემა ცალკეული i-ური სიმბოლო. დროის შედარებით გრძელ T მონაკვეთში შეიძლება გადაიცეს n=T/t სიმბოლო, ანუ n - თანმიმდევრობა სიმბოლოებისა:

$$v = (v_1,\dots, v_n) \in V_n \ , \qquad (1.3.1)$$

სადაც V_n ვექტორული სივრცეა GF(2) ველზე (სიმარტივისათვის განიხილება კოდირების ორობითი სისტემა. ზოგადად კოდირების სისტემას განიხილავენ GF(q) ველზე. (იხ.მე-2 თავი)) .

ფიზიკურად კავშირის არხში გადაიცემა გარკვეული სიმძლავრისა და ენერგიის იმპულსები. იმპულსის გადაცემას დროის t მონაკვეთში პირობითად შეესაბამება v_i=1, ხოლო პაუზას – v_i=0 სიმბოლო. დავუშვათ, რომ კავშირის არხში სიმბოლოს უშეცდომოდ გადაცემის ალბათობა არის q, შეცდომის ალბათობა _ p=1-q; p არის ალბათობა იმისა, რომ 1-ის გადაცემის შემთხვევაში არხის მეორე მხარეს მივიღებთ 0-ს ან 0-ის გადაცემისას –1-ს.

განიხილება დისკრეტული ორობითი სიმეტრიული არხი დამოუკიდებელი შეცდომებით, ანუ არხი მეხსიერების გარეშე, რომლის პრინციპული სქემა მოცემულია სურ.1.1-ზე:



სურ:1.1.

დავუბრუნდეთ სისტემის განუსაზღვრელობის საკითხს.

თავი 2
კრიფტოგრაფიული მეთოდების მოკლე მიმოხილვა

კრიპტოგრაფიას დიდი ხნის ისტორია აქვს. მას ჯერ კიდევ ცეზარის დროს იყენებდნენ, თუმცა მისი თეორიული საფუძვლები მხოლოდ 1948 წელს ჩამოყალიბა კლოდ შენონმა [3]. ინფორმაციის კრიპტოგრაფიული დაცვა ძირითადად გამოიყენებოდა სახელმწიფო სტრუქტურებში, სამხედრო საქმესა და დიპლომატიაში. ინფორმატიზაციის მზარდმა განვითარებამ მოითხოვა საბირჟო, საბანკო, კომერციული ინფორმაციის, იურიდიული დოკუმენტების, ავადმყოფობის ისტორიებისა და ბევრი კრიპტოგრაფიის სამთავრობო სტანდარტები.

რისთვის არის საჭირო ინფორმაციის დაცვა? რისი გაკეთება შეუძლია ინფორმაციის გამტაცებელს - ჰაკერს? მას შეუძლია შეცვალოს ინფორმაცია თავისი მიზნებისთვის, გაიფართოვოს თავისი კანონიერი უფლებამოსილებანი, გაიგოს, ვის რა ინფორმაციასთან აქვს შეხება, შეუშალოს ხელი მომხმარებლებს შორს ინფორმაციის გაცვლას. კრიპტოგრაფია იცავს ინფორმაციას, როგორც არასანქცირებული შეღწევებისაგან, ისე კომპიუტერული ვირუსისაგან.

ტერმინი „კრიფტოგრაფია“ მოიცავს სამეცნიერო-ტექნიკურ სფეროს, რომელიც იყოფა ორ ძირითად ნაწილად: კრიფტოგრაფიული სისტემის სინთეზი და კრიფტოანალიზი. პირველი ცდილობს, შექმნას ინფორმაციის დაშიფრა-დასაიდუმლოების მეთოდები, ხოლო კრიფტოანალიზი ცდილობს, გამოიკვლიოს ან „გატეხოს“ მეთოდები.

დაშიფვრა გულისხმობს ორი პროცედურის რეალიზაციას: ინფორმაციის საწყისი ღია ტექსტის დაშიფრა, დაშიფრული ტექსტის მიღების მიზნით; დაშიფრული ტექსტიდან საწყისი ტექსტის აღდგენა- დეშიფრაცია.

ორივე შემთხვევაში ადგილი აქვს ტექსტის გარდაქმნას განსაზღვრული ალგორითმის სიმრავლიდან, რომელიც ქმნის კრიპტოგრაფიული სისტემას: სისტემის ნაწილს, რომელიც ახორციელებს ინფორმაციული ტექსტის კონკრეტულ გარდაქნას, ეწოდება გასაღები. როგორც წესი (თუმცა, არა ყოველთვის), გასაღების სიგრძე გაცილებით ნაკლებია ტექსის სიგრძეზე.

თანამედროვე კრიპტოალგორითმები იყოფა ორ კლასად - სიმეტრიული და ასიმერტიული. შიფრაციის სიმეტრიულ სისტემებში[3] დასაშიფვრის გასაები (საიდუმლო) ძირითადად ემთხვევა დეშიფრაციის საიდუმლო გასაღებს, ხოლო შიფრაციის ასიმეტრიულ სქემებში (კრიპტოგრაფია ღია გასაღებით) დაშიფვრის ღია გასაღები არ ემთხვევა დეშიფრაციის საიდუმლო გასაღებს (აღსანიშნავია, რომ ინფორმაციის დაცვა, საზოგადოდ, კომპიუტერულ სისტემებში გაცილებით ფართოა ცნებაა, რასაც მხოლოდ [2] მონოგრაფიაც კი საკმაოდ ნათლად წარმოაჩენს).

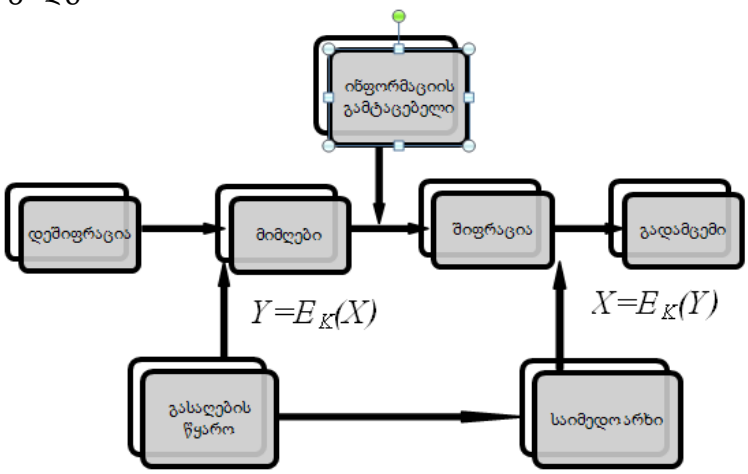
კრიპტოგრაფიაში მიღებულია კერკჰოფის წესი: შიფრის მედეგობა უნდა განისაზღვრებოდეს მხოლოდ გასაღების საიდუმლოობით. ინფორმაციის გამტაცებელს ან კრიპტოანალიტიკოსს

შეუძლია, იცოდეს ყველა მონაცემი, გარდა გასაღებისა. ითვლება, რომ კრიპტოსისტემა გახსნილია, თუ გამტაცებელს დასაშვებზე მეტი ალბათობით შეუძლია შემდეგი ოპერაციების ჩატარება; საიდუმლო გასაღების პოვნა, გარდაქმნის ეფექტური ალგორითმის შესრულება, რომელიც ფუნქციონალურად ექვივალენტურია საწყისი კრიპტოალგორითმისა.

იმისთვის, რომ კრიფტოსისტემა გახსნილად ჩაითვალოს, საჭიროა არა მხოლოდ გასაღების გახსნის (ანუ საიდუმლო გასაღების პარამეტრების მიღების) ალგორითმის ჩვენება, არამედ იმის ჩვენებაც, რომ ეს ალგორითმი შეიძლება შესრულდეს რეალურ დროში. ალგორითმის სირთულე ითვლება კრიპტოსქემის ერთ-ერთ მთავარ მახასიათებლად და მას კრიპტომედეგობა ეწოდება.

2.1. მეთოდები, რომლებიც ითვალისწინებს გასაღების საიმედო არხით გაცვლას

სიმეტრიული (შენონისეული) კრიპტოსისტემის კლასიკური მოდელი შეიძლება შემდეგნაირად წარმოვიდგინოთ:



სურ:1.1

ამ მოდელში სამი მონაწილეა: გადამცემი, მიმღები და ინფორმაციის გამტაცებელი. გადამცემის ამოცანაა, ღია არხით გადასცეს შეტყობინება დაცული ფორმით. ამისთვის ის საწყის X ტექსტს დაშიფრავს K გასაღებით და გასცემს დაშიფრულ Y ტექსტს. მიმღების ამოცანაა, გაშიფროს მიღებული Y ტექსტი და აღადგინოს X შეტყობინება. იგულისხმება, რომ გადამცემს აქვს გასაღბის საკუთარი წყარო და მის მიერ გენერირებულ გასაღებებს წინასწარ საიმედო არხოთ (სპეციალური კურიერი) გადასცემს მიმღებს. ინფორმაციის გამტაცებლის (ჰაკერის) ამოცანაა, წაიკითხოს გადაცემული შეტყობინებები.

არსებობს კრიპტოსისტემის მრავალი მეთოდი. მაგალითად, ცეზარის, ვიჟინერის, ვერნამის და სხვა, რომლებშიც გამოყენებულია x ინფორმაციის გარდასახვის სხვადასხვა ფუნქცია *y=f(x),* რომელთათვისაც არსებობს შებრუნებული f(y)=x ფუნქციები, რაც შეიძლება იყოს სიმბოლოთა ჩვეულებრივი გადანაცვლება, ჩასმა, მოდულით შეკრება-გამრავლების ოპერაციები და ა.შ [3].

ცეზარის მეთოდი: ანბანის ყოველ სიმბოლოს შეესაბამება გარკვეული რიცხვი, მაგალითად, ანბანში მისი ადგილის შესაბანისი ნომერი. ტექსტის ყოველ სიმბოლოს ემატება ფიქსირებული სიმბოლო (იკრიბება მათი შესაბამისი რიცხვები ფიქსირებული მოდულით, რომელიც ტოლია ანბანში სიმბოლოების რაოდენობისა) და იწერება მიღებული რიცხვის შესაბამისი სიმბოლო. მაგ:

ინფორმაცია: A B C D E

გასაღბი: D D D D D

კრიპტოგრამა: E F G H I

ვიჟინერის მეთოდი: მთელი ტექსტი იშიფრება მისი სიგრძის ტოლი შფრით. შიფრაცია ხდება ანალოგიურად. ეს არის გაუხსნელი მეთოდი, თუ ერთი შიფრი გამოიყენება მხოლოდ ერთხელ.

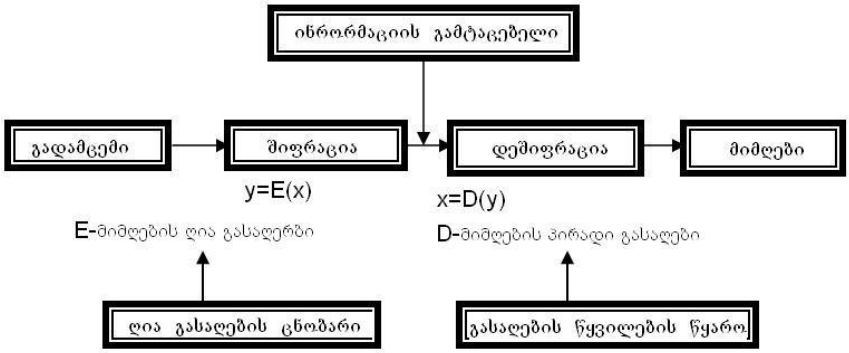
ამ მეთოდზე და ასევე, ჩანაცვლებებსა და გადანაცვლებებზე არის აღებული კლასიკური კრიპტოსისტემის სტანდარტი DES (Data Encryption Standart) [13]. იგი აგებულია ბლოკური კრიფტოალგორითმით (ბლოკის სიგრძეა 64 ბიტი) 64 ბიტის სიგრძის გასაღებით, რომლის პოზიციათაგან 56 შემთხვევითია, ხოლო 8 გამოიყენება გასაღბის კონტროლოსთვის. ეს ალგორითმი დამუშავებულია ფირმა IBM-ის მიერ და რეკომენდირებულია სტანდარტების ნაციონალური ბიუროს მიერ ეკონომიკის ღია სექტირებში გამოყენებისათვის. DES არის ძალიან ადვილად რეალიზებადი და სწრაფქმედებადი, მაგრამ მისი უმთავრესი ნაკლი არის ის, რომ გასაღების პერიოდულად შესაცვლელად საჭიროა საიდუმლო არხი (სპეციფიკური). როცა ქსელში n მომხმარებელია, საჭიროა $\frac{n(n-1)}{2}$ გასაღების გენერაცია და გაცვლა (რადგან ყოველ წყვილს თავისი შიფრი უნდა ჰქონდეს), რაც ძალიან რთულდება დიდი n-ის დროს.

2.2. გასაღების ღია არხით გაცვლის მეთოდები

არსებობს გასაღების სისტემის მართვის ამოცანის გადაჭრის და ღია არხით (საიდუმლო არხის გარეშე) კრიპტოგრაფიული კავშირის სხვადასხვა მეთოდი. ერთ-ერთი მათგანი დამყარებულია საიდუმლო გასაღების ღია გაცვლის იდეაზე. მისი არსის გასაგებად გამოვიყენოთ შემდეგი ანალოგია: ვთქვათ, არსებობს ორი ლექსიკონი: X-Y (მაგალითად ქართულ-ინგლისური), რომელიც ყველასთვის მისაწვდომია და Y-X (ინგლისურ-ქართული), რომელიც აქვს მხოლოდ ერთ პიროვნებას. ქართული ტექსტის ინგლსურად თარგმნა შეუძლია ყველას, ხოლო ინგლისური ტექსტის ქართულად, საჭიროა ქართულ-ინგლისური ლექსიკონის მთლიანი გადარჩვა, რაც ძალიან დიდ დროს მოითხოვს და პრაქტიკულად შეუძლებელია. ასევეა კრიპტოგრაფიაშიც. ყოველი მომხმარებელი ღაი არხში აცხადებს თვის X-Y ლექსიკონს, ხოლო Y-X ლექსიკონი აქვს მხოლოდ მას. ყველას შუძლია, გაუგზავნოს მას ინფორმაცია, ხოლო წაკითხვა მხოლოდ მას შეუძლია.

სწორედ ეს თვისებები უდევს საფუძვლად გასაღბების ღია განაწილების მეთოდებს, რომელთაც ასიმეტრიულ მეთოდებსა აწოდებენ. ამ მეთოდებშიც მაღალ კრიპტომედეგობას განაპირობებს

გამოთვლების დიდი სირთულე. ღია გასაღებების მოდელი შეგვიძლია შემდეგნაირად წარმოვიდგინოთ:



სურ.1.2.

სისტემის მოდელში სამი მონაწილეა: გადამცემი, მიმღები და ინფორმაციის გამტარებელი. გადამცემი X ტექსტს დაშიფრავს მიმღბის ღია E გასაღბით და გაგზავნის დაშიფრულ Y ტექსტს. მხოლოს მიმღებს შეუძლია Y ტექსტის გაშიფვრა და X-ის წაკითხვა, რადგან მხოლოდ მას აქვს საიდუმლოი D გასაღები. იგულისხმება, რომ გამგზავნს აქვს გასაღებების წინასწარ ან საიმედო არხით გადაცემემა გადამცემს. ასე რომ, N-აბონენტიან ქსელში ყოველი აბონენტი გამოიმუშავებს გასაღბების საკუთარ წყვილს (E,D) და აქვეყნებს E--ს. ე.ი. ქსელში იქნება შირაციის N ღია გასაღბები და დეშიფრაციის N საიდუმლო გასაღები. ეს ხსნის $\frac{N(N-1)}{2}$ გასაღბის საჭიროების პრობლემას.

2.2.1.დიფი-ჰელმან მერკლეს მეთოდი [14].

ალგორითმი ეყრდნობა გალუას GF(p) მარტივ ველში

ლოგარითმების გამოთვლის სირთულეს. ვთქვათ, $y = a^x mod\ p \quad (1 \le x \le q-1)$, სადაც a ველის პრიმიტიული ელემენტია. ამბობენ, რომ x არის y-ის ალგორითმი a ფუძით GF(p) ველში: $x = log_a Y \quad (1 \le y \le p-1)$ y-ის გამოთვლა x-ის მეშვეობით არ წარმოადგენს სირთულეს და მოითხოვს მაქსიმუმ $2 \cdot log_2 p$ - გამრავლების ოპერაციას, მაგალითად: $a^{18} = (((a^2)^2)^2)^2 \cdot a^2$ [15]. მეორე მხრივ, x-ის გამოთვლა y-ის მეშვეობით გაცილებით უფრო რთულია და მოცემული p-სთვის ცნობილი საუკეთესო ალგორითმების გამოყენებითაც კი $p^{1/2}$ ოპერაციათა რაოდენობას მოითხოვს [15].

გასაღბის გაცვლა ხორციელდება სქემით: ცნობილია ორი

მარტივი: p (მარტივი) და a (მთელი). $(1 \le X_i, X_j \le p-1)$.

i-ური მომხმარებელი გამოთვლის $y_{i=} a^{x_i} mod\ p$ რიცხვს და

ღია ფაილით უგზავნის მას j-ურ მომხმარებელს. თავის მხრივ, j-ური მომხმარებელი გამოითვლის $y_j = a^{x_j} mod\ p$ რიცხვს და ღიად უგზავნის მას i-ურ მომხმარებელს.

i-ური მომხმარებელი y_j-ის მეშვეობით აფორმებს

გასაღებს: $g_{ij} = y_j^{x_i} modp$

j-ური მომხმარებელი y_i-ის მეშვეობით აფორმებს

გასაღებს: $g_{ji} = y_i^{x_j} mod p$

ეს გასაღებები იდენტურია, რადგან:

$$y_j^{x_i} = (a^{x_j})^{x_i} = a^{x_j x_i} = (a^{x_i})^{x_j} = y_i^{x_j} (mod p)$$

ე.ი. $g_{ij} = g_{ji}$.

2.2.2.რაივესტ, შამირისა და ეიდელმანის ალგპრითმი- RSA.

განსხვავებით I ალგორითმისგან, RSA ახორციელებს

დაშიფრული ინფორმაციის ღის არხით გადაცემას. ალგორითმი ეყრდნობა ეილერის ცნობილ თეორემას: $x^{\varphi(N)} \equiv x mod N$, სადაც φ(N) ეილერის ფუნქციაა - N - ზე ნაკლები და მასთან ურთიერთ მარტივი რიცხვების რაოდენობა [17],$(1 \leq x \leq N)$.

საზოგადოდ, φ(N)-ის გამოთვლა დიდი რიცხვებისთვის

რთულია, მაგრამ ცნობილია, რომ $\varphi(N) = (p-1)(q-1)$, როდესაც N=pq, სადაც p და q ნებისმიერი მსრტივი რიცხვებია.

ალგორითმის განხორციელებისას მოიძებნება მაღალი

რიგის ორი მარტივი რიცხვი p და q, რომლებიც საიდუმლოდ ინახება, ხოლო N გამოცხადდება (ყველასთვის, Y და Z-ისთვის), საიდუმლოდ რჩება გამოთვლილი φ(N). შემდეგ მოიძებნება ისეთი e და d რიცხვები, რომ $ed = 1 mod \varphi(N)$, (სადაც e ურთიერთმარტივია φ(N) -თან) [15]. ამის შემდეგ გამოცხადდება.

ინფორმაცია გადაეცემა ისეთი $M_1, M_2 \ldots$ მთელი რიცხვების

მიმდევრობით, რომელთაგან თითოეული M მოთავსებულია [0, N-1] ინტერვალში. M ინფორმაციის დაშიფრა ხდება შემდეგნაირად: $C = M^e mod\ N$, სადაც C არის დაშიფრული ტექსტი. $ed = 1 mod \varphi(N)$ ანუ $ed = k \cdot \varphi(N) + 1$, რადგან $X^{k \cdot \varphi(N)+1} = X mod\ N$.

ნებისმიერი მთელი X-ისთვის [0,N-1] შუალედიდან და

ნებისმიერი k-სთვის დეშიფრაცია შეიძლება განხორციელდეს C-ს აყვანით d-ხარისხში: $C = M^{ed} = M^{k \cdot \varphi(N)+1} = M mod\ N$.

მიმღები აცხადებს e და N რიცხვებს, რომელთა

მეშვეობითაც გადამცემი დაშიფრავს ტექსტს, ხოლო გაშიფვრა შეუძლია მხოლოდ მიმღებს, რადგან მხოლოს მას აქვს საიდუმლო გასაღები- d რიცხვი. ჰაკერისთვის ძირითადი სირთულე მდგომარეობს φ(N)-ის გამოთვლაში, რადგან მისთვის უცნობია p და q რიცხვები, ხოლო დიდი მნიშვნელობის N-ისთვის φ(N)-ის გამოთვლა ძნელია. ალგორითმის მედეგობა ემყარება მთელი რიცხვების მამრავლებად დაშლის სირთულეს.

კრიპტოგრაფიული სისტემა შეგვიძლია შემდეგნაირადაც

წარმოვიდგინოთ: არის ორი მომხმარებელი: X და Y. X ირჩევს N_1, e_1 და d_1 რიცხვებს, ხოლო Y-N_2, e_2 და d_2-ს. e_1, N_1 და e_2, N_2 გამოცხადებულია (ცნობილია ყველასთვის). როდესაც X უგზავნის Y-ს

ინფორმაციას, გოგო ტექსტს დაშიფტავს ჯერ საკითსტი d_1 და N_1-ით, ხოლო შემდეგ- Y-ის e_2 და N_2-ით:

$$C_1 - M^{d_1} mod N_1, \quad C - C_q^{e_2} mod N_2 = (M^{d_1} mod N_1)^{e_2} mod N_2.$$

ტექსტის გასაშიფრად Y ჯერ გამოიყენებს საკუტარ d_2-ს და

აღადგენს C_1, ხოლო შემდეგ გამოიყენებს X-ის e_1-ს და აღადგენს M-ს:

$$C^{d_2} = C_1^{e_2 d_2} = C_1^{k \cdot \varphi(N_1)+1} = C_1 mod N_2$$

$$C_1^{e_1} = M^{d_1 e_1} = M mod N_1.$$

სხვა მომხმარებელი ვერ წაიკითხავს დაშიფრულ ტექსტს

იმის გამო, რომ არ იცის d_2(e_1 იცის), ხოლო ვერ გაგზავნის X-ის სახელით ყალბ ინფორმაციას იმიდ გამო, რომ არ იცის d_1, რომელიც საჭიროა დასაშიფრად.

2.2.3.ელგამალის ალგორითმი [16].

ალგორითმი გამოიყენება ღია ტექსტის ელექტრონული

ხელმოწერისთვის. ტექსტი არ დაიშიფრება, მაგრამ მას დაემატება ხელმოწერა (რომელიც ორობით კოდს წარმოადგენს) და მიმღები შეძლება ტექსტის ჭეშმარითების დადგენას. ძირითადი ტოლობაა: $a^s = a^{xR+kM} mod p$, სადაც M- ინფორმაციაა (ორობითი მიმდევრობა), p-მარტივი რიცხვი. მისი განზომლება განსაზღვრავს შემოწმების ღია კოდის განზომილებას. k-ერთჯერადი სემთხვევითი (ფსევდოშემთხვვითი) მთელი დადებითი რიცხვია, x- X-ის საიდუმლო გასაღები.

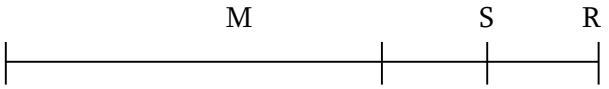
M-ტექსტის დაშივრა არ ხდება. მას მიეწერება S და R,

რომლებიც შემდეგნაირად გამოითვლება:

$R = a^k mod p, S = (xR + kM) mod (p-1)$, სადაც y, p და a

წინასწარაა ცნობილი (a მთელი დადებითი რიცხვია a>1): $y = a^x mod p$.

გადაცემული ინფორმაცია ღებულობს შემდეგ სახეს:



M||S||R (A||B||C წარმოადგენს A,B,C სიტყვების

თანმიმდევრულ შეთავსებებს- კონკატენციას.)

საზოგადოდ, მოცემული სიდიდეები მთელი დადებითი

რიცხვებია, წარმოდგენილი GF(2) ველზე განსაზღვრული ვექტორთა სახით.

Y-ისთვის (და Z-ისთვისაც) ცნობილია a,y,R,M და S.

Y ხელმოწერის სისწორეს ამოწმებს ტოლობით:

$$a^s = y^R \cdot R^M mod p$$

$$a^s = a^{xR+kM} = (a^x)^R \cdot (a^k)^M = y^R R^M (mod p)$$

Y-მა, ერთი მხრივ, a უნდა აახარისხოს S ხარისხად a^S,

მეორე მხრივ, უნდა გამოთვალოს $y^R.R^M$. თუ შესრულდა ზემოაღნიშნული პირობა, ე.ი. ხელმოწერა სწორია.

ცხრილში მოცემულია DES და RSA

კრიპტოალგორითმების მახასიათებლების შედარება.

მახასიათებელი	**DES**	**RSA**
შიფრაციის სიჩქარე	მაღალი	დაბალი
გამოყენებული სიჩქარე	გადანაცვლება და ჩასმა	ხარისხში აყვანა
გასაღების სიგრძე	56 ბიტი	500 ბიტზე მეტი
კრიპტოანალიზის სირთულე (იგი განსაზღვრავს ალგორითმის მედეგობას)	გასაღების სივრცეში მთლიანი გასარჩევა	მამრავლებად დაშლა
გასაღების გენერაციის დრო	მილიწამები	წუთები
გასაღების ტიპი	სიმეტრიული	ასიმეტრიული

ამ ცხრილიდან ჩანს, რომ DES ალგორითმი ბევრად უფრო

სწრაფქმედი, მოხერხებული და ადვილად რეალიზებადია, ვიდრე RSA. RSA-ს გამოყენების რთულდება ალგორითმი, მაგრამ წყდება ღია არხით ინფორმაციის გადაცემის პრობლემა.

თავი 3

კოდირების ალგორითმი (შეცდომამედეგი)

3.1.სისტემური სტრუქტურა

კოდირების თეორია (შეცდომების გამსწორებელი

კოდური სისტემები), როგორის შესავალში იყო აღნიშნული, ინფორმაციის თეორიაში ფუნდამენტული შრომების [1-5] შედეგად ჩაისახა და მის ერთ-ერთ ძირითად სამეცნიერო-ტექნიკურ მიმართულებას წარმოადგენს. თეორიული თვალსაზრისით იგი ეყრდნობა მათემატიკის ისეთ დარგს, როგორიცაა ალგებრა, რიცხვთა თეორია, კომბინატორიკა, გრაფთა თეორია და კომბინატორული ალგორითმები.

ძნელია თანამედროვე დიდი ინფორმაციული

კომპიუტერული სისტემების დასახელება (ინტერნეტი, ქსელური თუ სხვა ავტომატური მიზნობრივი სისტემები), რომელთა იერარქიის გარკვეული საფეხურები არ მოიცავენ კოდირების სტრუქტურებს და სისტემებს. მათდამი ინტერესი და მნიშვნელობა მხოლოდ ზოგიერთი ნაშრომისა და მონოგრაფიის დასახელებაც აშკარად ჩანს [1,3-9,19].

კოდირების ალგებრულ თეორიაში განსაკუთრებული

მნიშვნელობა აქვს ვექტორულ ქვესივრცეთა ბაზისური მატრიცების სტრუქტურულ თვისებებს. მათ გამოყენებას ეფუძვნება ორიგინალური მეთოდის სინთეზი, რომელიც განხილულია ნაშრომის III თავში. მოქმედებანი GF(q) სასრულ ველზე განსაზღვრულ $V_{n,q}$ სივრცეში (კერძოდ ვექტორული და მატრიცული გარდაქმნები) წარმოადგენებ თანამედროვე ინფორმაციული სისტემების მოდელირების ერთ-ერთ მნიშვნელოვან შემადგრნრლ რგოლს. ამასთან, ნაშრომის III თავში განხილული კრიპტოგრაფიული სისტემა ძირითადად ეფუძნება ზემოაღნიშნულ გარდაქმნებს. ამდენად, II თავში მოცემულია კოდირების ზოგიერთი ალგორითმული სტრუქტურის თვისებები.

ინფორმაციული სიტყვა

$$a = (a_1, a_2, a_3 \dots a_n) \in V_n \tag{2.1}$$

არის n- განზომილებიანი ვექტორი; V_n სიმრავლე სალიას

GF(q) ველზე განსაზღვრული n- განზომილებიანი ვექტორული სივრცეა;

$q=p^m$ -მარტივი რიცხვის ხარისხი (ემდგომში ზოგადობის დაურღვევლად შეიძლება მივიჩნიოთ, რომ q=2).

განვიხილოთ ვექტორულ სივრცეთა ზოგიერთი თვისება.

ყოველი V_n ვექტორული სივრცისთვის არსებობს სივრცის ბაზისი- წრფივად დამოუკიდებელ $b^{(i)} = (b_1^{(i)}, \dots b_n^{(n)}) \in V_n$ ვექტორთა $\{b^{(1)},\dots b^{(n)}\}$ (i=1, . . . n) სიმრავლე, რომლის სტრიქონთა კომბინაციებით $a = a_1 b^{(1)} + \dots + a_n b^{(n)} \in V_n$ (2.2)

მიიღება V_n სივრცე ; აქ $a_i(i=1, \dots n)$ GF(2) ველის

ელემენტია.

ვთქვათ, V ? V_n არის V_n სივრცის k განზომილების

ქვესისტემა. მაშინ არსებობს V ქვესივრცის ბაზისური მარტიცა, $G = \begin{bmatrix} g_{11}\, g_{12} & \cdots & g_{1n} \\ g_{21}\, g_{22} & \cdots & g_{2n} \\ \dots\ \dots & & \\ g_{k1}\, g_{k2} & \dots & g_{kn} \end{bmatrix}$

(2.3)

რომლის სტრიქონთა სივრცე (სტრიქონთა წრფივი კომპინაციები) k განზომილების ქვესივრცეა. V ქვესივრცის ნულვანი V′ სივრცე r = n – k განზომილებისაა, რომლის ბაზისი არის

$$G = \begin{bmatrix} h_{11}\, h_{12} & \cdots & h_{1n} \\ h_{21}\, h_{22} & \cdots & h_{2n} \\ \dots\ \dots & & \\ h_{r1}\, h_{r2} & \dots & h_{rn} \end{bmatrix} \tag{2.4}$$

ე.ი. სამართლიანია ტოლობა:

$$GH^T=0. \tag{2.5}$$

სადაც H^T არის H მატრიცის ტრანსპონირებული მატრიცა. (2.5)- გამოსახულებიდან გამომდინარეობს, რომ ნებისმიერი $a \in V$

$$aH^T=0. \tag{2.6}$$

ამასთან, ნებისმიერი b ∈ V′ ვექტორისათვის

$$bG^T=0. \qquad (2.7)$$

სადაც G^T არის G მატრიცის ტრანსპონირებული მატრიცი. ე.ი. V′ ქვესივრცის ნულოვანი სივრცე არის V :

$$HG^T=0. \qquad (2.8)$$

<u>**განსაზღვრება 2.1.**</u> k განზომილების V ქვესივრცეს

ეწოდება წრფივი (n,k)-კოდი, თუ მისთვის G და H, შესაბამისად, მაწარმოებელი და შემომაწმებელი მატრიცებია.

V კოდისთვის H მატრიცის შემამოწმებელი ეწოდება, რადგან (2.6) გამოსახულება არის ნებისმიერი a∈V ვექტორის V ქვესიმრავლისთვის მიკუთვნების აუცილებელი და საკმარისი პირობა.

იმ შემთხვევაში, თუ:

$$aH^T\neq 0. \qquad (2.8')$$

უნდა მივიჩნიოთ, რომ $a \notin ?V$. საზოგადოდ:

$$aH^T=S. \qquad (2.9)$$

სადაც $S=(S1, \ldots S_r)$-r განზომილების ვექტორი, ანუ სინდრომია (მაშასადამე, თუ S=0, a∈V და თუ S≠0, $a \notin ?V$).

განვიხილოთ V_n/V ფაქტორჯგუფი. (2.9) პირობა

ცალსახად განსაზღვრავს {a} ფიქსირებული მოსაზღვრე კლასის ვექტორების სიმრავლეს, რადგან ყოველ მათგანს და მხოლოდ მათ V_n/V ფაქტორჯგუფებში, (2.9) პირობის თანახმად, შეესაბამება ერთი და იგივე სინდრომი, ე.ი. თუ a′∈{b} და b′∈{b} სხვადასხვა მოსაზღვრე კლასების ნებისმიერი ელემენდებია, მაშინ:

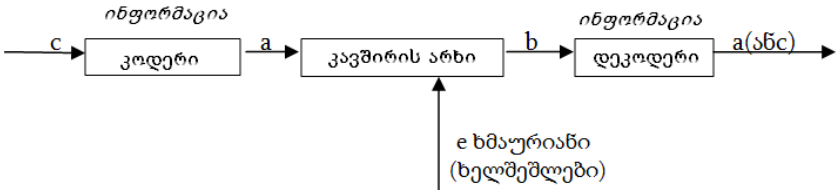
$$a'H^T\neq b'H^T \qquad (2.10)$$

H მატრიცის გარკვეული სტრუქტურის აგებით

განისასღვრება V კოდი და განხორციელდება საჭირო V_n/V პაქტორიზაცია, ე.ი. მატრიცი შეიძლება წარმოადგენდეს ინფორმაციული სისტემების შემადგენელ ნაწილს.

განვიხილოთ კოდირების სისტემის (სიგნალების

გადაცემის) მატრიცული სქემა (სურ. 2.1).



სურ. 2.1

სისტემის შესვლაზე ინფორმაციის წყაროდან მიეწოდება k

განზომილების ინფორმაციული ვექტორი, ანუ შეტყობინება - $c-(c_1, \ldots c_k)$. კოდი c ვექტორს გარდაქმნის $a=(a_1, \ldots a_n) \in V$ ვექტორად, რომელსაც შეუძლია გაასწოროს შეცდომები.

არხში მოქმედი ხმაურის (ხელშეშლების)ზეგავლენით

მოსალოდნელია c ვექტორის სახეცვლილება (დამახინჯება), რის შედეგად არხის მეორე ბოლოზე ვღებულობთ

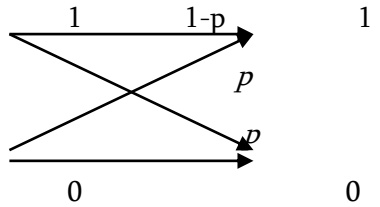
$$b=a+e \tag{2.11}$$

ვექტორს, სადაც $b=(b_1, \ldots b_n)$ არის ვექტოროი $b=(a_1+e_1, \ldots a_n+e_n)$,

ხოლო $e=(e_1, \ldots e_n)$ შეცდომის ვექტორიაქ. შეკრების ოპერაცია განხილულია იმ ველში, რომელზედაც განსაზღვრულია a და e ვექტორები (ორობით შემთხვევაში იგულისხმება GF(2) ველი).

განვიხილოთ ინფორმაციის გადაცემის პროცესი ხელშეშლის

შემთხვევაში. სქემატურად ორობითი სიმეტრიული არხიმოცემულია სურ. 2.2-ზე.



სურ.2.2.

ორობითი სიმეტრიული არხისთვის განსაზღვრავენ სიმბოლოს

(ანუ a ვექტორის $a_i \in GF(2)$ (i=1, . . . n) კომპონენტს) დამახინჯების p ალბათობას. ალბათობა იმიას რომ დეკოდერის შესავალზე მიღებული სიმბოლო თანხვდება გადაცემული სიმბოლოს, q=1-p სიდიდის ტოლია; ამრიგად, კავშირის სიმეტრიული არხი, ორობითი თანმიმდევრობით (ანუ ვექტორით) შესავალსა და გამოსავალზე, ისეთი არხია, რომლის ყოველი a_i სიმბოლო მიმღებ მხარეზე მიღება უცვლელად (დაუმახინჯებლად) ფიქსირებული 1-p ალბათოვით და სახეს იცვლის (მახინჯდება) P ალბათობით. კავშირისა ასეთარხს უეწოდებენ არხს მეხსიარების გარეშე ან არხს დამოუკიდებელი შეცდომებით. ცხადია, (2.6) და (2.11) - დან გამომდინარეობს, რომ

$$bH^T=eH^T=s \tag{2.11'}$$

<u>**განსაზღვრება 2.2.**</u> a (2.1) ვექტორის w(a) წონა a ვექტორია

არანულოვანი კომპონენტების რაოდენობის ტოლია. GF(2) ველზე განსაზღვრული ვექტორია წონა (ორობითი შემთხვევა)

$$W(a) = \sum_{i=1}^{n} a_i \tag{2.12}$$

<u>**განსაზღვრება 2.3.**</u> ჰემინგის d(a,b) მანძილი a და b ვექტორებს

შორის იმ ერთსახელა კომპონენტების (a_i,b_i) წყვილების რაოდენობაა, რომელთათვისაც $a_i \neq b_i$ (i=1, . . . n). GF(2) ველზე განსაზღვრული ვექტორისთვის (ორობითი შემთხვევა)

$$d(a,b) = \sum_{i-1}^{n} (a_i + b_i) \ . \tag{2.13}$$

შევნიშნოთ რომ (2.13) გამოსახულების (a_i,b_i) წყვილებისთვია

განსაზღვრულია შეკრებია ოპერაცია GF(2) ველში. ამიტომ აქ და შემდგომში, სადაც ეს არ იწვევს გაუგებრობას, არ ვიხმართ მოდულით 2 შეკრების აღმნიშველ სიმბოლოს.

<u>განსაზღვრება 2.4.</u> ვთქვათ $V \subset V_n$ ქვესივრცის ნებისმიერი a,b $\in V$ წყვილისთვის d(a,b) ჰემინგის მანზილია, მაშინ

$$D(V)=\min d(a,b)\ ; \quad a \neq b \in V \tag{2.14}$$

სიდიდე არის V ქვესივრცის მინიმალური მანძილი.

დავუშვათ, e (2.11) გამოსახულებაში $w(e)\leq t$ წონის ვექტორია:

$$e=b-a \tag{2.15}$$

ე.ი. a სიტყვის არხში გადაცემის შედეგად ადგილი აქვს t - ჯერად შეცდომას.

შეცდომების გამასწორებელი კოდების თეორიის

ძირითადი პრობლემა მრავალჯერადი შეცდომებისათვის შეიძლება ჩამოყალიბდეს ორი განსხვავებული სახით: 1. საჭიროა, აიგოს გარკვეული ფიქსირებული სიმძლავრის v კოდური სიმრავლე, რომლისთვისაც ყოველი მიღებული b=a+e სიტყვის შესაბამისი $a\in V$ სიტყვის აღდგენა შესალებელი იქნება e (2.15) შეცდომის შემთხვევაში t რიცხვის რაც შეიძლება მაღალი მნიშვნელობისთვის. 2. კავშირის არხში ფიქსირებული $\leq t$ - ჯერადი e (2.15) შეცდომების არსებობის პირობებში საჭიროა, აიგოს რაც შესაძლებელი მაღალი სიმძლავრის კიდური სიმრავლე V, რომლისთვისაც შესაძლებელი იქნება ყოველი მიღებული b=a+e სიტყვის აღდგენა. თუ $V \subset ?V_n$ კოდური სიმრავლე დასმულ ამოცანას წყვეტს, მაშინ V ქვესივრცის t-ჯერადი შეცდონის გამასწორებელი წრფივი კოდი ეწოდება. V კოდურ სიმრავლეს ახასიათებს: n=k+r კოდური ვექტორებისსიგრძე, ანუ კოდური (2.1) სიტყის სიმბოლოების რაოდენობა, k - ინფორმაციულ სიმბოლოთა რაოდენობა, ანუ კოდის მაწარმოებელი (2.3) მატრიცის რანგის მნიშვნელობა, r - შემოწმებულ სიმბოლოთა რაოდენობა, ანუ კოდის შემამოწმებელი (2.4) მატრიცის რანგის მნიშვნელობა, (ასეთ კოდს t -ჯერადი შეცდომების გამასწორებელ (მაკორექტირებელ) წრფივ (n,k) - კოდს უწოდებენ).

თუ K (2.1) სიტყვათა რაიმე სიმრავლეა, მაშინ t-ჯერადი

შეცდომების გასასწორებლად საჭიროა, შესრულდეს შემდეგი პირობა: $d(K)\geq 2t+1$ (2.16)

მართლაც, (2.16) გამოსახულება ნიშნავს იმას ,რომ $a+b\in K$

ნებისმიერ წყვილს შორის მანძილი არ არის 2t+1 სიდიდეზე ნაკლები და მაშასადამე, t - ჯერადი შეცდომა შეიძლება გასწორებულ იქნას.

შეცდოების კორექტირების საკითხი წრივ V კოდში

შეიძლება ასე გადაწყდეს:

იმისთვია რომ $V \subset V_n$ კოდი ასწორებდეს t- ჯერად შეცდომებს,

საჭიროა რომ

$$W(a)\geq 2t+1 \tag{2.17}$$

ნებისმიერი a∈ V არანულოვანი ვექტორისთვის.

მართლაც თუ

$$w_{min}(V) = \min_{a \neq 0 \in V} w(a) = 2t + 1 \ , \tag{2.18}$$

მაშინ ნებისმიერი a≠b∈V წყვილისთვის

$$d(V)=d(a,b)=w(a+b)\geq 2t+1 \tag{2.19}$$

რადგან, თუ დავიშვებთ, რომ

$$w(a+b)<2t+1, \tag{2.20}$$

მაშინ, v ქვესივრცის შეკრების ოპერაციის მიმართ ჩაკეტილობის გამო, მივიღეთ, რომ

$$w(c)<2t+1 \tag{2.21}$$

სადაც a+b=c∈V. (2.21) გამოსახულება ეწინაარმდეგება (2.17) დაშვებას. მაშასადამე, V კოდის მინიმალური მანძილი არ არის 2t+1 სოდოდეზე ნაკლები (როდესაც სრულდება (2.17) პირობა), ე.ი.

$$d(V)\geq 2t+1 \ , \tag{2.22}$$

და კოდი ასწორებს t-ჯერად შეცდომებს.

ამრიგად, ვექტორთა მინიმალური წონისა და

მინიმალური მანძილის გამოყენებით შეიძლება აიგოს დამოუკიდებელი t- ჯერადობის შეცდომების გამასწორებელი კოდი, მაგრამ (2.16) და (2.17) პირობების შუალო შესწავლა დაკავშირებულია ვექტორების გადასინჯვასთან 2^k (საზოგადოდ q^k) სიმძლავრის V კოდურ სიმრავლეშებში. ეს კი კომპიუტერული რეალიზაცვიის დროს n- ისა და k-ს დიდი მნიშვნელობისთვის იწვევს პრაქტიკულად გადაულახავ სიქნელეებს. მდგომარეობიდან გამოსვლას წარმოადგენს საჭირო სტრუქტურის G (2.3) და H (2.4) მატრიცების აგება.

ვიდრე კოდების მატრიცულ აღწერას განვიხილავთ,

ვისარგებლოთ შემდეგი ცნობილი განსაზღვტებით [20]:

<u>განსაზღვრება 2.5.</u> GF(q) ველზე განსაზღვრული V_n ვექტორული სივრცის $a^{(1)}$ . . . $a^{(w)}$ ვექტორების

ერთობლიობას ეწოდება წრფივად დამოკიდებული, თუ

$$a^{(1)} a^{(1)} +. . . + a^{(w)} a^{(w)} =0 \tag{2.23}$$

$a^{(1)}$ ∈GF(q)(i=1, . . . w) სკალარების გარკვეული

მნისვნელობისთვის, როდესაც $a^{(1)}$, . . . $a^{(w)}$) ელემენტები ერთდროულად არ უდრის ნულს. $a^{(1)}$, . . . $a^{(w)}$ ვექტორების ერთობლიობა წრფივად დამოკიდებულია, თუ ის წრფივად დამოკიდებული არ არის.

არსებობს კავშირი (2.23) წრფივად დამოკიდებულებისა

და (2.2) წრფივ კომბინაციებს შორის. მართლაც, თუ $a^{(1)}$, . . . $a^{(w)}$ ვექტორთა რაიმე ერთობლიობის რომელიმე ვექროტი არის დანარჩენი ვექტორებივ წრფივი კომბინაცია, მაშინ ეს ერთობლიობა

წრფივად დამოკიდებულია. (2.6) პირობიდან გამომდინარე, H მატრიცის იმ ვექტორ-სვეტის ჯამი, რომლებიც შეესაბამებიან a ვექტორის არანულოვან კომპონენტებს (2.6′) გამოსახულებაში, სწრაფად დამოკიდებულ ერთობლიობას შეადგენენ.

კოდების მატრიცული აღწერა გაცილებით უფრო

კომპაქტურია V სიმრავლესთან შედარებით. კოდის საჭირო სტრუქტურის აგების თვალსაზრისით მნიშვნელოვანია შემდეგი ცნობილი თეორემა [80], რომელიც, როგორც შემდეგ პარაგრაფში არის ნაჩვენები, ახალ მნიშვნელობას იძენს m- პათეტიკური შეცდომების განხილვასთან დაკავშირებით.

<u>**თეორემა 2.1.**</u> V წრფივი კოდის w წონის წოველ კიდურ a

ვექტორს V ქვესივრცის ნულოვანი სივრცის H მატრიცაში შეესაბამება w რაოდემობის გარკვეული სვეტების წრფივი დამოკიდებულება და, პირიქით, ყოველ წრფივ დამოკიდებულებას, შედგენილს w სვეტისგან, V სიმრავლეში შეესაბამება w წონის ერთი გარკვეული კოდური ვექტორი.

თეორემის მტკიცება ეკრდნობა იმ ფაქტს, რომ $a=(a_1, \ldots a_n)$

[2.1] ვექტორი არის კოდური მაშინ და მხოლოდ მაშინ, როდესაც ის აკმაყოფილებს [2.6] დამოკიდებულებას. ეს უკანასკნელი შეიძლება ასე გადავწეროთ:

$$\sum_{j=1}^{n} a_j h^{(j)} = 0 \qquad (2.24)$$

სადაც $h^{(j)}=(h_1^{(j)}, \ldots h_n^{(j)})$ არის H მატრიცის j- ური ვექტორ-

სვეტი. [2.24] წრფივ დამოკიდებულებაში მონაწილეობს ის ვექტორ-სვეტები, რომლებიც მასში შედის არანლოვანი a_j კოეფიციენტებით. ამრიგად, განსაზღვრება 2.2.-ის თანახმად, წეფივ დამოკიდებულებაში მონაწილეობს იმდენი ვექტორ-სვეტი, რა წონაცაა კოდური ვექტორი. მეორე მხრივ, w რაოდენობის $h^{(j)}$ ($j \in \{1,\ldots n\}$) ვექტორ- სვეტებისგან შედგენილ წრფივ დამოკიდებულებაში a ვექტორი შედის w რაოდენობის a_j არანულოვანი კოეფიციენტით. ამიტომ w რაოდენობის ვექტორ-სვეტების ყოველ წრფივ დამოკიდებულებას შეესაბამება w წონის a კოდური ვექტორი.

როგორც თეორემა 2.1. - დან ჩანს, მნშვნელოვანია H

მატრიცის სტრუქტურებისკვლევა V სივრცის საჭირო თვისებების მისაღებად. 2.1. თეორიიდან გამომდინარეობს შემდეგი შედეგი:

<u>**შედეგი 2.1**</u> V კოდის მინიმალური წონა არის w მაშინ და

მხოლოდ მაშინ, როდესაც მიდი ნულოვანი სივრცის ბაზისური H მატრიცის ნებისმიერი w-1 და ნაკლები რაოდენობის სვეტებისგან შემდგარი ერთობლიობა სრფივად დამოკიდებულია.

ამრიგად, T-ჯერადი შეცდომების გამასწორებელი V

კოდისთვის სამართლიანია შემდეგი გამოსახულება:

$$e_{2t}H^T \neq 0, \qquad (2.25)$$

სადაც e_{2t} 2t-ჯერადი შეცდომის ვექტორია.

2.1. შედეგი საშუალებას იძლევა ავაგოთ კოდები,

რომლებიც ორობით სიმეტრიულ არხში ასწორებენ t-ჯერად დამოუკიდებელ შეცდომებს.

თავი 4

ალგებრულ (მატრიცულ) სტრუქტურებზე დაფუძნებული კრიპტოგრაფიული მეთოდის სინთეზი

4.1 ზოგადი მიდგომა მატრიცული გასაღების მისაღებად

განვიხილოთმატრიცული გასაღების მეთოდი. ეს მეთოდი

მდგომარეობს შემდეგში: ტექსტი, ანუ a (2.1) ორობითი სიტყვა იყოფა გარკვეული სიგრძის ბლოკებად. შიფრაცია ხდება შემდეგნაირად: ბლოკი, როგორც x ვექტორი, გამრავლდება შესაბამისი განზომილების მატრიცაზე. დეშიფრაციის დროს კი მიღებული ვექტორი უნდა გავამრავლოთ შებრუნებულ მატრიცაზე, რაც აღადგენს საწყის ვექტორს:

$$x \cdot A = x' \;; \qquad x' \cdot A^{-1} = x \tag{3.1}$$

A მატრიცას მოეთხოვება, რომ იყოს გადაუგვარებელი

(ე.ი. დეტერმინანტი 0-ის ტოლი არ უნდა იყოს), რათა გააჩნდეს შებრუნებული, ე.ი. დგას პრობლემა გადაუგვარებელი მატრიცების სინთეზისა და მათი შებრუნებულის პოვნისა. როდესაც საქმე გვაქვს დიდი განზომილების მქონე მატრიცასთან, მათი შებრუნებულის გამოთვლა ცნობილი მეთოდით გარკვეულ დროს მოითხოვს. ამიტომ საჭიროა, შეიქმნას რეგულარული მეთოდები, რომლებიც მარტივი ალგორითმებით მოგვცემენ მატრიცის შებრუნებულს. ე.ი. გამოიყოს გარკვეული კლასი მატრიცებისა, რომელთა შებრუნებულის პოვნა რეალიზდება (გარკვეული წესით) რთული ალგორითმული გამოთვლის გარეშე.

განვიხილოთ ცნობილი მეთოდები მატრიცის

შებრუნებულის პოვნისას. ცნობილია არაერთი მეთოდი. შესაძლებელია $A=(a_{ij})^n$ მატრიცის შებრუნებულის პოვნა (თუ ის არასინგულარულია) შემდეგი სახით [11]:

$$A^{-1} = \begin{bmatrix} \frac{A_{11}}{|A|} & \frac{A_{21}}{|A|} \cdots & \frac{A_{n1}}{|A|} \\ \frac{A_{12}}{|A|} & \frac{A_{22}}{|A|} \cdots & \frac{A_{n2}}{|A|} \\ \cdots & \cdots & \cdots \\ \frac{A_{1n}}{|A|} & \frac{A_{2n}}{|A|} \cdots & \frac{A_{nm}}{|A|} \end{bmatrix} \tag{3.1}$$

სადაც A_{ij} არის A მატრიცის a_{ij} ელემენტის ალგებრული დამატება.

როგორც ვხედავთ, მიუხედავად იმისა, რომ ორობითი

GF(2) ველზე (3.1) ოპერაციების ჩატარება შედარებით ადვილია, საჭირო გამოთვლები, ცხადია, დროს მოითხოვს და, რაც მთავარია, მისი მეშვეობით ძნელია გარკვეული კლასის ფორმირება, რომლისთვისაც (3.1) მატრიცები მიიღება გრივიალური გზით.

დასახული მიზნის მიღწევა, ჩვენი აზრით, არც შედეგი

მატრიცული ნამრავლის მეშვეობითაა შესაძლებელი:

$$E_k\, E_{k-1} \cdots E_1\, A = 1, \tag{3.2}$$

და

$$E_k E_{k-1} \dots E_1 = A^{-1}, \tag{3.3}$$

სადაც A^{-1} A, მატრიცის მარცხენა შებრუნებული

მატრიცაა; $E_1 \dots E_k$ წარმოადგენენ ელემენტარულ მატრიცებს, რომელთა მეშვეობით A მატრიცი შესაძლებელია დავიყვანოთ კანონიკურ და, მაშასადამე, ერთეულოვან სახამდე.

განსხვავებულ მეთოდს წარმოადგენს A^{-1} მატრიცის i-ური სვეტის $x_1, x_2, \dots x_n$ ელემენტების მისაღებად (გამომდინარე $AA^{-1}=I$ ტოლობიდან) განტოლებათა შეგეგი სისტემის ამოხსნა:

$$a_{i1}x_1 + a_{k2}x_2 + \dots + a_{kn}x_n = \begin{cases} 0, \text{თუ } k \neq i \\ 0, \text{თუ } k = i \end{cases} \tag{3.4}$$

სადაც k=1,2,…n. რადგან $|A| \neq 0$, ამიტომ (3.4) სისტემას

აქვს ერთადერთი ამონახსნი, რის შედეგადაც, საზოგადოდ, A^{-1} შებრუნებული მატრიცი მიიღება.

კოდირეის ალგებრული თეორიიდან ცნობილია, რომ

მრავალწევრთა ალგებრაში GF(q) ველზე მოდულით f(x) მიიღება კლასები მატრიცებსა, რომლებიც წარმოქმნიან (2.3) და (2.4) სახის მაწარმოებელ და შემამოწმებელ ბაზისურ მატრიცებს, რომლებიც აკმაყოფილებენ (2.5) პირობას. შესაბამისი მატრიცების სტრიქონთა სივრცე წარმოადგენს მრავალწევრთა იდეალებს. ასეთი მატრიცებისათვის დამახასიათებელია g(x) და h(x) (g(x)·h(x)=f(x))მწარმოებელი მრავალწევრები, რომლებიც G (2.3) და (2.4) ბაზისური მატრიცების სტრიქონებს აფორმირებენ [8] (რაც განხილულია 3.2).

აღნიშნულის ანალოგიურად n რიგის კვადრატული

მატრიცები და მათი შებრუნებულები შესაძლოა შავწეროთ შემდეგი სახით:

$$A = \begin{bmatrix} a_1 & a_1 & a_3 & \cdots & a_{n-1} & a_n \\ 0 & a_1 & a_2 & \cdots & a_{n-2} & a_{n-1} \\ 0 & 0 & a_1 & \cdots & a_{n-3} & a_{n-2} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & & a_1 & a_2 \\ 0 & 0 & 0 & \cdots & 0 & a_1 \end{bmatrix}, \tag{3.5}$$

სადაც (3.5) მატრიცის სტქირონებს მრავალწევრთა იდეალების ბაზისური მატრიცების მსგავსად შეადგენენ GF(2) ველზე განსაზღვრული $a \in V_n$ ვექტორის კომპონენტები (ანუ A მატრიცს აწარმოებს $g=(a_1, \dots a_n)$ ვექტორის კომპონენტებით).

ფიქსირებული ვექტორისთვის შესაძლებელია

შებრუნებულის პოვნის ერთ-ერთი მეთოდით ((3.4) სისტემის ეშვეობით) განსაზღვროს h ვექტორის სახე ნებისმიერი მთელი დადებითი n-ისათვის. მაგალითად, (3.4)-ისა და მათემატიკური ინდუქციის გამოყენებით შეიძლება ვაჩვენოთ, რომ n რიგის

$$A = \begin{bmatrix} 1 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & 1 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & \dots & 1 & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix} \qquad (3.6)\leq$$

მატრიცისთვის, რომლის მაწარმოებელი ვექტორი არის

$g = (a_1, a_2, \dots\ a_n)\big(a_i = 1\ \text{თუ}\ i \leq 2\ \text{და}\ a_i = 0, \text{თუ}\ i > 2\big)$, შებრუნბულ მატრიცს აქვს სახე:

$$A^{-1} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 & 1 \\ 0 & 1 & 1 & \dots & 1 & 1 \\ 0 & 0 & 1 & \dots & 1 & 1 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & \dots & 1 & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix}, \qquad (3.7)$$

სადაც $h=a_1, \dots a_n)$, $a_i=1(1\leq i\leq n)$.

ანალოგიურად, $g=(a_1,a_2, \dots a_n)$ $(a_1=1, i\leq 3; a_i=0, i>3)$ და

$h=(a_1,a_2, \dots a_n)$ $(a_i=1\ , i=3k+1, i=3k+2; a_i=0, i=3k)$ მაწარმოებელი ვექტორისთვის შესაბამისად მიიღება:

$$A = \begin{bmatrix} 1 & 1 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & 1 & 1 & \dots & 0 & 0 \\ 0 & 0 & 1 & 1 & \dots & 0 & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 & 0 \\ . & . & . & . & . & . & . \\ 0 & 0 & 0 & 0 & \dots & 1 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix}, \quad A^{-1} = \begin{bmatrix} 1 & 1 & 0 & \dots & 1 & 1 & 0 \\ 0 & 1 & 1 & \dots & 0 & 1 & 1 \\ 0 & 0 & 1 & \dots & 1 & 0 & 1 \\ 0 & 0 & 0 & \dots & 1 & 1 & 0 \\ . & . & . & . & . & . & . \\ 0 & 0 & 0 & \dots & 0 & 1 & 1 \\ 0 & 0 & 0 & \dots & 0 & 0 & 1 \end{bmatrix} \qquad (3.8)$$

ე.ი (3.8) მატრიცებს აწარმოებს შესაბამისად g და h

ვექტორები იმ პირობით, რომ მატრიცებში ყოველი i-ური სტრიქონი წარმოადგენს (i-1)-ე სტრიქონის კომპონენტების გადანაცვლებას ერთი პოზიციით და რომ (3.8) მატრიცის ყოველი (i+1)-ე სტრიქონის პირველი i კომპონენტი ნულის ტოლია.

მოცემული (3.6) – (3.8) სახის მატრიცების სინთეზის

მიზანი მდგომარეობს იმაში, რომ არ განვახორციელოთ მატრიცის შებრუნებულის გამოთვლა, არამედ გარკვეული მარტივი შესაბამისობით შეგვეძლოს შებრუნებულის პოვნა.

იმისთვის, რომ გასაღებს გააჩნდეს მაღალი მედეგობა,

საჭიროა, სიმრავლეში იყოს საკმაოდ დიდი რაოდენობის გასაღები. ავიღოთ n-განზომილებიანი ერთ-ერთი განხილული მატრიცა და მისი შებრუნებული. თუ საწყის მატრიცაში გადავაადგილებთ სტრიქონებს, მიღებული მატრიცის შებრუნებულის საპოვნელად საჭიროა, საწყისი მატრიცის შებრუნებულში გადავაადგილოთ შესაბამისი სიტყვები, ე.ი. ერთი მატრიციდან შეგვიძლია მივიღოთ n! რაოდენობის მატრიცა. ასევე შესაბამისი სვეტებისა და სტრიქონების გადანაცვლებით მიიღება ისეთივე რაოდენობის მატრიცა და მთლიანობაში $(n!)^2$ სიმრავლე ფიქსირებული g და h ვექტორებისათვის.

მატრიცული გასაღების მეთოდით საჭირო გასაღების

სინთეზი შესრულდება შემდეგი თანმიმდევრობით: შეირჩევა გარკვეული g და h ვექტორები, მოხდება A da A^{-1}

მატრიცების გენერაცია დაშემდეგ შემთხვევითი (ფსევდოშემთხვევითი) რიცხვის შესაბამისად მატრიცებში განხორციელდება სათანადო გადანაცვლებები.

რა უპირატესობა გააჩნია მატრიცულ მეთოდს ვიჟინერის

მეთოდთან შედარებით, სადაც ტექსტი ასევე ბლოკებად იყოფა? ვიჟინერის მეთოდში რომ „გატყდეს“ ერთ-ერთი ბლოკის ნაწილი, ცნობილი გახდება თვითონ გასაღების ნაწილი და, შესაბამისად, ეს ნაწილი „გატყდება“ ყოველ ინფორმაციულ ბლოკში; ხოლო მატრიცული გასაღების დროს ერთ ბლოკში ინფორმაციის გახსნით შეუძლებელია მატრიცის პარამეტრების მიღება და მთლიანად გასაღების „გატეხვა“.

ნაშრომში მატრიცული გასაღებების მეთოდი

გამიზნულია კომბინირებული კრიპტოსისტემის შესაქმნელად, რომელშიც ერთად იქნება გამოყენებული ღია არხით სარგებლობის ცნიბილი მეთოდები და მატრიცული მეთოდი, თუმცა გარკვეული მომხმარებლისათვის მისი არც ცალკე (სხვა მეთოდებისგან დამოუკიდებლად) გამოყენებაა მიღებული.

4.2.კრიპტოგრაფიული გასაღებების სინთეზი მრავალწევრთა ალგებრაში

განხილული ამოცანის გადაწყვეტა უფრო

მიზანდასახული იქნება, თუ გამოვიყენებთ კოდირების ალგებრულ სტრუქტურებს, კერძოდ, GF(q) ველზე (სიმარტივისთვის განვიხილავთ GF(2)ველს) მოდულით f(x) მრავალწევრთა ალგებრაში იდეალების თვისებებს.

ცნობილია, რომ n-განზომილებიან მრავალწევრთა ნაშთთა კლასები მოდულით f(x) GF(2) ველზე წარმოქმნიან მრავალწევრთა A_n ალგებრას და, მაშასადამე, ვექტორულ V_n სივრცეს (ბგულისხმობთ, რომ

$$a=(a1,\ldots an) \in Vn \text{ და } a(x) = \sum_{i=0}^{n} a_i x^i \in A_n$$

წარმოადგენენ ექვივალენტურ ელემენტებს).

ცნობილია, ასევე, რომ A_n ალგებრაში ნებისმიერი I იდეალისთვის არსებობს ერთადერთი ნორმალური g(x) მრავალწევრი მინიმალური ხარისხისა ისეთი, რომ {g(x)} ნაშთთა კლასი ეკუთვნის I იდეალს და პირიქით, თითოეული ნორმირებული g(x) მრავალწევრი, გამყოფი g(x)-ისა, აწარმოებს გარკვეულ I იდეალს, რომელშიც g(x) არის მინიმალური ხარისხის მრავალწევრი ისეთი, რომ g(x) ნაშთთა კლასი ეკუთვლის I იდეალს.

სამართლიანია შემდეგი

<u>თეორემა 3.1.</u> ვთქვაქთ,f(x)=g(x)h(x), სადაც f(x) არის n ხარისხის მრავალწევრი, ხოლო h(x)-k ხარისხისა. მაშინ {g(x)} ნაშთთ კლასით ნაწარმოები იდეალი მოდულით f(x) მრავალწევრთა ალგებრაში არის კი განზომილებისა.

ეს ნიშნავს, რომ g(x) მრაცალწევრის ხარისხი არის

$$n-k=r \tag{3.9}$$

სამართლიანია აგრეთვე

<u>თეორია 3.2.</u> ვთქვათ, f(x)=g(x) და h(x) ნორმირებული მრავალწევრებია და f(x)=g(x)h(x), მაშინ {a(x)} ნაშთთა კლასი ეკუთვმის h(x)-ით ნაწარმოები იდეალის ნულოვან სივრცეს მაშინ და მხოლოდ მაშინ, როდესაც ის ეკუთვნის g(x) -ით ნაწარმოებ იდეალს.

ზემოთქმულიდან გამომდინარეობს შემდეგი

<u>შედეგი 3.1.</u> ვთქვათ, f(x)= g(x)h(x), სადაც f(x)-n ხარისხის და g(x)-r ხარისხის მრავალწევრებია. მაშინ

$$GH^T=0,$$

სადაც G და H მატრიცებს შესაბამისად g(x) და h(x) მრავალწევრები აწარმოებენ.

$g=(g_0, \dots g_{n-1})$ ვექტორის კომპონენტების ციკლური გადანაცვლება i პოზიციით წარმოადგენს $g(i)=(g_i \dots g_{n-1})$ ვექტორს; ანუ $g(x)=1+x+ \dots + x^r$ მრავალწევრის i-ურო გადანაცვლება გვაძლებს $g(x^{(i)})=x^i g(x) \mathrm{mod}(x^n-1)$ მრავალწევრს.

ვთქვაქთ g(x)h(x)=x^n-, 1 g(x) და h(x) აწარმოებენ შესაბამისად I და I' იდეალებს. მაშინ

$$G=\begin{bmatrix} g_0\, g_1 \dots & g_r & 0 \dots & 0 \dots 0 \\ 0 & g_0 \dots & g_{r-1}\, g_r \dots & 0 \dots 0 \\ \dots\dots\dots & \dots\dots\dots & & \dots\dots\dots \\ 0 & 0 \dots & 0 \quad 0 \dots & g_0 \dots g_r \end{bmatrix}, \tag{3.10}$$

$$H=\begin{bmatrix} h_0^* & h_1^* \dots & h_k^* & 0 \dots & 0 \dots 0 \\ 0 & h_o^* \dots & h_{k-1}^*\, h_k^* \dots & & 0 \dots 0 \\ \dots\dots\dots & & \dots\dots\dots & & \dots\dots\dots \\ 0 & 0 \dots & 0 \quad 0 \dots & & h_0^* \dots h_k^* \end{bmatrix} \tag{3.11}$$

რაც ნიშნავს, რომ ნებისმიერი $g(x^{(i)})$ და $h(x^{(j)})$

მრავალწევრებისთვის სამართლიანია ტოლობა:

$$g(x^{(i)})\, h(x^{(j)}) \equiv 0 \ \mathrm{mod}(x^n-1) \tag{3.12}$$

სადაც I,j∈ {1,…n}. თუ გავითვალისწინებთ, რომ GF(2)

ველზე მრავალწევრთა და ვექტორთა ნამრავლი არ ემთხვევა ერთმანეთს, ნაშინ ნებისმიერი g∈I ვექტორისთვის

$$gH^{*T}=0, \tag{3.13}$$

სადაც H^* მატრიცი ნაწარმოებია h^* ვექტორით, რომელიც

შეიცავს h ვექტორის კომპონენტებს, ჩაწერილს საწინააღმდეგო თანმიმდევრობით (ე.ი.h^* წარმოადგენს h ვექტორის სარკისებულ შებრუნებულს).

მაშასადამე, (3.12) და (3.13) ექვივალენტური ტოლობები

(რაც ჩვენთვის მნიშვნელოვანია) სამართლიანია, რადგან I და I* იდეალები წარმოადგენენ ჩაკეტილ სიმრავლეებს ვექტორთა ნებისმიერი ციკლური წანაცვლების მიმართ.

განვიხილოთ (3.5) მატრიცის შესაბამისი n რიგის

კვადრატული მატრიცები, ნაწარმოები g(x) და h(x) მრავალწევრებით (რომელთა კოეფიციენტების მეშვეობით მიღებულია (3.10) და (3.11) მატრიცების სტრიქონები):

$$A_1 = \begin{bmatrix} g_0 & g_1 & \dots & g_r & 0 & \dots & 0 \\ 0 & g_0 & \dots & g_{r-1} & g_r & \dots & 0 \\ . & . & . & . & . & . & . \\ 0 & 0 & \dots & 0 & 0 & \dots & g_0 \end{bmatrix}, \tag{3.14}$$

$$A_2 = \begin{bmatrix} h_0 & h_1 & \dots & h_k & 0 & \dots & 0 \\ 0 & h_0 & \dots & h_{k-1} & h_k & \dots & 0 \\ . & . & . & . & . & . & . \\ 0 & 0 & \dots & 0 & 0 & \dots & h_0 \end{bmatrix}, \tag{3.14'}$$

სადაც A_2 მატრიცის ყოველი j-ური სვეტი წარმოადგენს h'(j) ვექტორს მოდულით x^n-1 ალგებრაში, რომლის i-ური კონპონენტები იგივეა, რაც $h^*(x)x^{r+j-1}$ ვექტორის კონპონენტები, თუ i≤j და h'_i=0, თუ i>j. ყოველივე ზემოთქმულიდან ((3.9) პირობის გათვალისწინებით) გამომდინარეობს რომ

$$g(i)h'(j)^T = \begin{cases} 0, & i \neq j \\ 1, & i \neq j \end{cases}. \tag{3.15}$$

სადაც h'^T ვექტორი (3.15) ტოლობაში წარმოადგენს ვექტორ-სვეტს (ანუ h' ვექტორის ტრანსპონირებულ ვექტორს)

მაშასადამე, სამართლიანია

<u>თეორემა 3.1.</u> ვთქვათ, g(x) და h(x), შესაბამისად, r და k ხარისხის მრავალწევრებია GF(2) ველზე მოდლით (x^n-1) ალგებრაში ისეთი, რომ g(x)h(x)=x^n-1, ხოლო A_1 და A_2 n რიგის g(x) და h(x) მრავალწევრებით ნაწარმოები მატრიცაა (3.14) . მაშინ A_1 და A_2 ურთიერთშებრუნებულია, ე.ი.

$$A_1 A_2 = I, \quad A_2 A_1 = I,$$

სადაც I ერთეულოვანი მატრიცაა.

არსებობს კონსტრუქციული მეთოდი x^n-1 მოდულით ალგებრაში g(x) და h(x) მრავალწევრების მიღებისა, რომელთათვისაც g(x)h(x)=x^n-1. რაც **<u>3.1.</u>** თეორემით მიღებული მეთოდის კონსტინუქციული განხორციელებისთვის საჭირო წინაპირობებს უზრუნველყოფს.

<u>მაგალითი.</u> მრავალწევრთა A_7 ალგებრაში მოდულით x^7-1 GF(2) ველზე g(x)=1+x+x^3 და h(x)=1+x+x^2+x^4 მრავალწევრებისათვის g(x)h(x)=x^7-1. მიიღება ურთიერთშებრუნებული მატრიცები.

$$A_1 = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \; A_2 = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.16}$$

მე-4 თავში განხილულია 3.1 თეორემით მიღებილი მატრიცების ფორმირებისთვის ალგორითმები და კომბინირებული კრიპტოგრაფიული სისტემის მოდელირების საკითხები.

თავი 5

კრიპტოგრაფიული მეთოდის სინთეზი. მეთოდის ალგორითმული და პროგრამული განხორციელება

5.1. კრიპტოგრაფიული სისტემის აღწერა

გადამცემი დამიმღები ახდენენ გასაღების სინთეზს, რომელიც იმუშავებს გარკვეული დროის მანძილზე (რამდენიმე დღიდან რამდენიმე თვემდე - მედეგობის მიხედვით). ამ ხნის განმავლოვაში ყველა ინფორმაცის იშიფრება ამ გასაღებით. გასაღები წარმოადგენს n-განზომილებიან მატრიცას და მის შებრუნებულს. მისი მედეგობა დამოკიდებულია n რიცხვის სიდიდეზე.

გასაღების სინთეზი ხდება შემდეგნაირად: შექმნილია ბაზა, რომელშიც ჩაწერილია სხვადასხვა n-თვის g(x) და h(x) მრავალწევრები GF(2) ველიდან ისეთი, რომ g(x)h(x)=0 mod f(x), სადაც $f(x)=x^n-1$. g(x) და h(x) მრავალწევრების კოეფიციენტები წარმოადგენენ შესაბამისად g და h ვექტორების კომპონენტებს. ამ ვექტორებიდან მიიღება მატრიცები, რომელთა პირველი სტრიქონები წარმოადგენენ მოცემულ ვექტორებს, ხოლო ყოველი შედაგი სტრიქონი მიღებულია წინა სტრიქონის ერთი ნაბიჯით მარჯვნივ წანაცვლებით (მარცხნიდან ემატება 0).

გადამცემი დამიმღები ერთობლივად - გასაღებების ღია არხით გაცვლის რომელიმე მეთოდით - ახდენენ 3 რიცხვის სინთეზს: ერთი მათგანი მოახდენს ბაზიდან g(x) და h(x) მრავალწევრების შერჩებას, დანარჩენი ორის მიხედვით კი შეირჩება 1,2, ... n რიცხვების გარკვეული გადანახვლება: შემდეგ ერთი გადანაცვლების შესაბამისად დამშიფრავ მატრიცაში მოდხა სტრიქონების გადაადგილება, ხოლო გამშიფრავ მატრიცაში - სვეტების, მეორე გადანაცვლების შესაბამისად კი მიღებულ დამშიფრავ მატრიცაში მოხდება სვეტების გადანაცვლება, ხოლო გამშიფრავ მატრიცაში - სტრიქონების. მიიღება G დამშიფრავი და H გამშიფრავი მატრიცები.

გადამცემს გასაგზავნი ინფორმაცია გადაყავს ორობით მიმდებრობაში, რომელსაც ყოფს გასაღების (მატრიცის) განზომილების ტოლ ბლოკებად. შემდეგ თტოეულ ბლოკს მატრიცულად ამრავლებს G მატრიცაზე და მიღებულ ორობით მიმდებრობას უგზავნის მიიმღებს. მიმღები ამ მიმდევრობას ყოფს მატრიცის განზომილების ტოლ ბლოკებად და თითოეულ ბლოკს მატრიცულად ამრავლებს H მატრიცაზე. მიიღება საწყისი მიმდევრობა, რომლის მიხედვითაც იგი აღადგენს საწყის ტექსტს.

5.2. პროგრამული რეალიზაცია ეგმ-ზე და მაგალითი ექსპარიმენტისთვის

მოდელი შედგება სამი ნაწილისაგან; 1. გასაღების სინთეზი; 2. შეტანილი ინფორმაციის დაშიფრა; 3. დაშიფრული ინფორმაციიდან საწყისი ინფორმაციის აღდგენა. მოდელისთვის (როგორც კერძო მაგალითი) აღებულია 8- განზომილებიანი მატრიცა, ხოლო ტექსტი შეიცავს 5 სიმბოლოს. (ასო-ნიშანს).

1. გასაღების სინთეზი.

თავიდან მოცემული გვაქვს ორი 8-განზომილებიანი ვექტორი, რომლებიც წარმოადგენენ g(x) და h(x) მრავალწევრების კოეფიციენტებს(ან ჩვენ შეგვიძლია ეს კოეფიციენტები თვითონ შევიტყოთ). ამ ვექტორების კომპონენტების მიხედვით იქმნება მატრიცები. ჩვენ შეგვყავს რიცხვი 1-დან 8!=40320 ფარგლებში. ამ რიცხვის მეშვეობით ხდება 1,2, ... n რიცხვების გარკვეული გადანაცვლების სინთეზი. ამ გადანაცვლების შესაბამისად პირველ მატრიცაში ხდება სტრიქონების გადანაცვლება, ხოლო მეორეში-სვეტების. ასე მიიღება G დამშიფრავი და H გამშიფრავი მატრიცები.

2. ინფორმაციის (ტექსტის) დასიფვრა.

მეორე ნაწილში ჩვენ შეგვყავს ტექსტი, რომელიც უნდა დაიშიფროს (5 სიმბოლო). ჯერ ხდება ამ სიმბოლოების ASCII-კოდებში გადაყვანა, ხოლო შემდეგ- კოდების 8-თანრიგიან ორობით რიცხვებად გადაქცევა. თითოეული რიცხვი, როგორც ორობითი ვექტორი, მატრიცულად გამრავლდება G დამშიფრავ მატრიცაზე. მიიღება დაშიფრული ორობითი ვექტორი. ხდება მათი, როგორც ორობითი რიცხვების, გადაყვანა ათობითში, ხოლო შემდეგ ათობითი რიცხვების, როგორც ASCII-კოდების, შესაბამისი სიმბოლოების მიღება. მიიღება დაშიფრული ტექსტი.

3. საწყისი ტექსტის აღდგენა.

მესამე ნაწილში ხდება დაშიფრული ტექსტიდან საწყისი ხექსტის აღდგენა. დაშიფრული ტექსტის სიმბოლოები გადაიყვანება ASCII-კოდებში, როგორც ვექტორები, მატრიცულად გამრავლდება H გამშიფრავ მატრიცაზე, მიღებული ორობითი რიცხვები გადაიყენება ათობითში და დაიწერება მათი შესაბამისი სიმბოლოები. აღდგება საწყისი ტექსტი.

მოცემული m რიცხვის საშუალებით 1,2, ... n რიცხვების გარკვეული გადანაცვლების იდენტიფიკაცია

m რიცხვების ფორმირება ხდება გადამცემისა და მიმღების მიერ გასაღებების ღია გაცვლის რომელიმე მეთოდით (მოდელში m რიცხვი კლავიატურიდან შეგვყავს).

1,2, ... n რიცხვების გადანაცვლებათა რაოდენობა არის n გადანაცვლებები შეიძლება დალაგდეს მიმდევრობით, ანუ გადაინომროს (დამყარდეს ურთიერთცალსახა შესაბამისობა 1,2, ... n! რისცვებსადა 1,2, ... n რიცხვების გადანაცვლებებს შორის).

მაგალითად: 1,2,3; 1,3,2; 2,1,3; 2,3,1; 3,1,2; 3,2,1 (თითოეული გადანაცვლება წარმოვადგინოთ ათობით რიცხვებად (123) და დავალაგოთ ზრდადობის მიხედვით).

ჩვენი მიზანია, მოცემული m რიცხვების საშუალებით მოვახდინოთ იმ გადანაცვლების ფორმირება, რომელიც მას შეესაბამება. ეს ხორციელდება შემდეგნაირად:

ჩავთვალოთ, რომ m მოთავსებულია 1-სა და n!-ს შორის (წინააღმდეგ შეთხვევაში m-ს ავიღებთ n!-ის მოდულით (0-ის ნაცვლად n!-ს ავიღებთ)).

[1,n!] შუალედი დავყოთ n ბლოკებად (მონაკვეთებად),რომელთაგან თითოეულის სიგრძეა (n-1)! (n(n-1)!=n!). ვნახოთ, რომელ ბლოკში ხვდება მოცემული m რიცხვი დაამ ბლოკის ნომერი დავწეროთ პირველ ადგილზე. შემდეგ დავამყაროთ შესაბამისობა 1,2, ... n-1 მიმდევროვისას და ზრდადობის მიხედვით დალაგებულ დარჩენილ (n-1) რიცხვს შორის (დარჩენილი (n-1) რიცხვი გადავნომროთ). ვნახოთ mრიცხვის ნომერი თავის ბლოკში დაამ რიცხვს დავარქვათ m_1. ეს ბლოკი კიდევ დავყოთ (n-1) ბლოკებად, რომელთაგან თითოეულის სიგრძეა (n-2)! ((n-1)·(n-2)!=(n-1)!). ვნახოთ, რომელ ბლოკში ხვდება m_1 რიცხვი და ამ ბლოკის ნომრის შესაბამისი რიცხვი (გადანომრილი (n-1) რიცხვიდან) დავწეროთ მეორე ადგილზე. შემდეგ დარჩენილი (n-2) რიცხვი დავალაგოთ ზრდადობის მიხედვით და გადავნომროთ და ა.შ. სანამ არ დავწერთ ყველა რიცხვს. მიიღება m რიცხვის შედაბამისი გადანაცვლება.

ორობითი მიმდევრობის მიხედვით 1,2, ... n რიცხვების გარკვეული გადანაცვლების ფორმირება

თუ n რიცხვი დიდია, 1,2, ... n რიცხვების გადანაცვლების იდენტიფიკაცია ზემოთ მოყვანილი მეთოდით სირთულეებთანაა დაკავშირებული, რადგან მასში ოპერაციები ხორციელდება n! რიგის რიცხვებზე. ამის გამო, შესაძლებელია, უფრო მიზანშეწონილი იყოს ორობითი მიმდევრობის მიხედვით პირდაპირ 1,2, ... n რიცხვების გადანაცვლების ფორმირება. ეს შეიძლება შემდეგნაირად განხორციელდეს: გადამცემი და მიმღები ახდენენ ორობით მიმდევრობის ფორმირებას. ეს მიმდევრობა იყოფა თანაბარი სიგრძის ბლოკებად. ბლოკის სიგრძე ისე უნდა იყოს შერჩეული, რომ უდიდესი ორობითი რიცხვი 11...1 (რომლის თანრიგი ტოლი იქნება ბლოკის სიგრძის), გადაყვანილი ათობითში, მეტი ან ტოლი იყოს n-ის. გადანაცვლების ფორმირება ხდება შემდეგნაირად: ვიღებთ პირველ ბლოკს, გადაგვყავს ათობითში (თუ მეტია n-ზე, ვიღებთ n-ის მოდულით) და ვწერთ პირველ ადგილზე. ყოველი შედეგი რიცხვის აღებისას იმ შემთხვევაში, თუ ის უკვე იყო გამოყენებული, ვზრდით (ან ვამცირებთ) მას 1-ით და ისევ ვადარებთ გამოყენებულ რიცხვებს. თუ ისიც იყო გამოყენებული, ისევ 1-ით ვზრდით (ან ვამცირებთ) და ა.შ. სანამ არ მივიღებთ ისეთ რიცხვს, რომელიც გამოყენებული არ ყოფილა. ამრიგად შესაძლებელია [1;n] შუალედის ყოველი რიცხვის მიღება, ანუ H მატრიცის სტრიქონების გადანომვრა და შესაბამისად მისი ფორმირება.

თავი 6
სიმეტრიული სისტემის მეთოდები, პრაქტიკული სამუშაოებისათვის

6.1 ცეზარის მეთოდი

ცეზარის მეთოდი, რომელიც აგრეთვე ცნობილია შემდეგი სახელწოდებებით: ცეზარის შიფრი, ცეზარის ალგორითმი, დაძვრის შიფრი, პრაქტიკული გამოყენების მიზნით ძალზე მარტივია. მიუხედავად ამისა, იგი საკმაოდ ცნობილია დაშიფვრა/გაშიფვრის სხვა მეთოდებთან შედარებით.

ცეზარის ალგორითმის გამოყენების არსი, რომელშიც იგი იყენებდა მხოლოდ ლათინურ ალფავიტს, მდგომარეობს შემდეგში: დასაშიფრი TI-ის S_i (i=1,2,…,z) სიმბოლო ჩაენაცვლება K (K=3) პოზიციით მისგან მარჯვნივ - R_k (მარცხნივ - L_k), დაშორებული D_i-იური სიმბოლოთი. ამგვარი წესით მიღებული შიფროტექსტი რა თქმა უნდა არ არის მდგრადი, ადვილად შეიძლება მისი „გატეხვა“ და პრაქტიკული თვალსაზრისით არავითარ ღირებულებას არ წარმოადგენს. თუმცა უნდა აღინიშნოს, რომ ცეზარის მიერ შემოთავაზებული მეთოდის ძირითადი იდეა წარმატებით გამოიყენება კრიპტოგრაფიის შედარებით რთულ სისტემებში, მაგალითად, როგორიცაა ვიჟინერისა და ვერნამის შიფრები და სხვ.

როგორც ცნობილია, ცეზარი მის მიერ შედგენილ წერილებში იყენებდა დაშიფვრა/გაშიფვრის, მარცხნიდან/მარჯვნივ და პირიქით მარჯვნიდან/მარცხნივ, შემდეგ სისტემას: მან ალფავიტიდან გამოყო სამი სიმბოლო (X,Y,Z) და გამოიყენა K=3 პოზიციით ჩანაცვლების შემდეგი მარტივი წესი (იგულისხმება,რომ დაშიფვრა ხორციელდება მარჯვნიდან მარცხნივ, ხოლო გაშიფვრა მარცხნიდან მარჯვნივ):

დაშიფვრის შემთხვევაში:	**გაშიფვრის შემთხვევაში:**
თუ $S_i \cong X$ მაშინ $D_i \cong A$	თუ $D_i \cong A$ მაშინ $S_i \cong X$
თუ $S_i \cong Y$ მაშინ $D_i \cong B$	თუ $D_i \cong B$ მაშინ $S_i \cong Y$
თუ $S_i \cong Z$ მაშინ $D_i \cong C$	თუ $D_i \cong C$ მაშინ $S_i \cong Z$

სადაც, $\cong$ - აღნიშნავს მის მარცხნივ მდგომი სიმბოლოს ცანაცვლებას ძისგან მარჯვნივ მდგომი სიმბოლოთი.

აღწერილიდან გამომდინარე, ზოგადად შემოთავაზებული ჩანაცვლება აღიწერება შემდეგი სახით:

$$X \leftrightarrow A;\ \ Y \leftrightarrow B;\ \ Z \leftrightarrow C \qquad (1)$$

დანარჩენ შესაძლო შემთხვევებში D_i განისაზღვრება K=3 პოზიციით მისგან მარჯვნივ მდებარე სიმბოლოთი. ანალოგიურად, თუ დაშიფვრა/გაშიფვრის პროცედურებს განვახორციელებთ მარჯვნიდან მარცხნივ ჩანაწერი (1)-ს მიირებს შემდეგ სახეს:

$$A \leftrightarrow X;\ \ B \leftrightarrow Y;\ \ C \leftrightarrow Z \qquad (2)$$

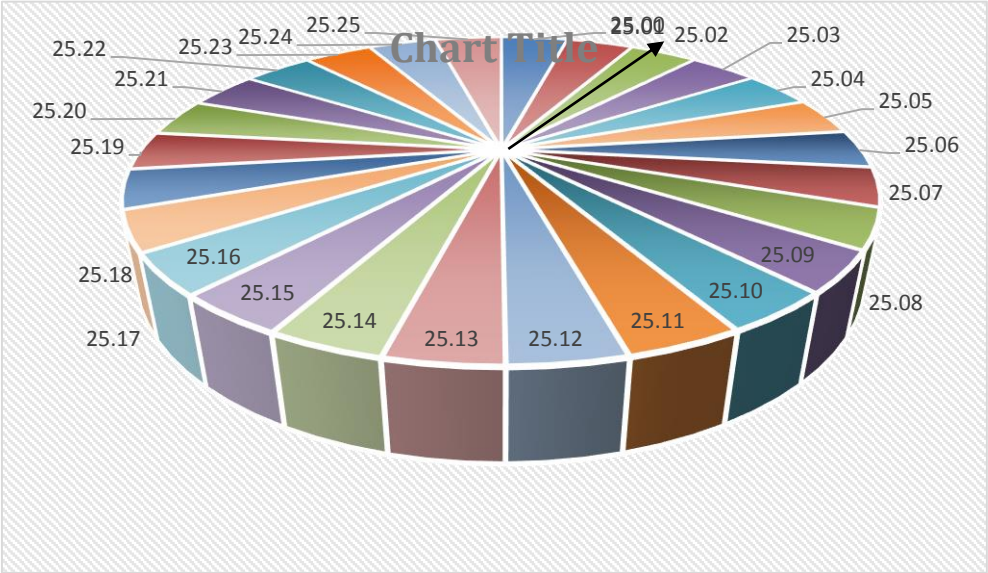
განვიხილოთ ცეზარის მიერ შემოთავაზებული მეთოდის რეალიზაციის პროცედურული შესრულების ორი ვარიანტი.

ვარიანტი 1. ვთქვათ, დასაშიფრია და შემდგომ გასაშიფრია ტექსტური ინფორმაცია - TI = „ XZ UNIVERSIT ZY”. დაშიფვრა/გაშიფვრის პროცესის ვიზუალური თვალსაჩინოებისათვის ვისარგებლოთ ცხრილით -ცხრ. 1, რომელშიც ლათინური ალფაბიტის სიმბოლოებს მინიჭებული აქვთ შესაბამისი რიგითი ნომრები mTel ricxvTa სიმრავლედან - Z_0 .

ცხრ. 1

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

TI დაშიფვრა (გაშიფვრა) შევასრულოთ სიმბოლოების K=3 პოზიციით გადანაცვლებით მარჯვნიდან (მარცხნიდან) - მარცხნივ (მარჯვნივ). ეტაპობრივი შესრულება დაშიფვრის პროცედურისა ნაჩვენებია ცხრ. 2.



ნახ. 1. ლათინური ანბანის ბორბალი.

ცხრ. 2

№	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
TI	X	Z		U	N	I	V	E	R	S	I	T	Y		Z	Y
S_i რიგითი № ცხრ. 1	A	C		20	13	8	21	4	17	18	8	19	B		C	B
D_i რიგითი № ცხრ. 1	A	C		23	16	11	24	7	20	21	11	22	B		C	B
შიფროტექსტი	A	C		X	Q	L	Y	H	U	V	L	W	B		C	B

შევნიშნოთ, რომ ცხრ. 2_ის 1,2,13,15,16 სვეტებში შეტანილი TI-იის სიმბოლოების ჩანაცვლება ხორციელდება (1) სტრიქონში ნაჩვენები წესით, ხოლო 4-12 სვეტებში შეტანილი TI-იის სიმბოლოების ჩანაცვლება კი ჩვეულებრივი წესით (K=3 სიმბოლოთი გადანაცვლების გზით), როგორც ეს აღწერილი იყო ზევით.

გაშიფვრის პროცედურული ნაბიჯები ანალოგიურია დაშიფვრის აღწერილი პროცედურული ნაბიჯებისა იმ განსხვავებით, რომ ჩასანაცვლებელი სიმბოლოები განისაზღვრება K=3 პოზიციით საათის ისრის მოძრაობის საწინააღმდეგო მიმართულებით (მარჯვნიდან მარცხნივ) გადანაცვლებით (იხ. ცხრ. 3).

ცხრ. 3

№	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
შიფროტექსტი	A	C		X	Q	L	Y	H	U	V	L	W	B		C	B
D_i რიგითი № ცხრ	X	Z		23	16	11	24	7	20	21	11	22	Y		Z	Y
S_i რიგითი № ცხრ	X	Z		20	13	8	21	4	17	18	8	19	Y		Z	Y
TI	X	Z		U	N	I	V	E	R	S	I	T	Y		Z	Y

ვარიანტი 2. წარმოვიდგინოთ ბორბალი („იღბლიანი“) ან საათის ციფერბლატი (ნახ. 1).

დავყოთ იგი n ტოლ ნაწილად (სექტორად), სადაც n ალფავიტის სიმძლავრეა(ჩვენს შემთხვევაში n=26). ყოველ ნაწილში მარჯვნიდან მარცხნივ მიმდევრობით ჩავწეროთ ალფავიტის თითო სიმბოლო: A,B,C,…,Y,Z. ფიგურის ცენტრში დავამაგროთ რადიუსის ტოლი მოძრავი ისარი. რიცხვების წილადი ნაწილი.00-.25 შეესაბამება ალფაბიტში სიმბოლოების რიგითი ნომრებს (იხ. ცხრ. 1).

TI_იის დასაშიფრავად, ავაყენოთ ისარი S_i - სიმბოლოზე და მივაბრუნოთ იგი საათის ისრის მიმართულებით K - დანაყოფით. ისრის დაფიქსირებული ახალი მდგომარეობა მიუთითებს სიმბოლოზე - D_i (შიფროტექსტის i-იურ სიმბოლოზე). შიფროტექსტის გასაშიფრად საკმარისია

ჩატარდეს აღწერილი პროცედურების საწინააღმდეგო მოქმედებები. კერძოდ, დავაყენოთ ისარი D_i - იურ სიმბოლოზე და მოვაბრუნოთ იგი საათის ისრის საწინააღმდეგო მოძრაობის მიმართულებით, K - დანაყოფით დაფიქსირებული ისრის მდგომარეობა მიუთითებს S_i - სიმბოლოზე.

6.2 ვიჟინერის მეთოდი

უნდა აღინიშნოს, რომ ისტორიული მონაცემებით, რაც დღესათვის არის ცნობილი კრიპტოგრაფიაში ბ. ვიჟინერის მეთოდის სახელწოდებით, ჯერ კიდევ 1467 წ. იყო შემოთავაზებული ლეონ ბატისტა ალბერტის მიერ, ხოლო 1518 წ. იაგან ტრესემუსმა ნაშრომში „პოლიგრაფია“ აღწერა თავის მიერ გამოგონებული მრავალალფავიტური ცხრილი ე.წ. tabula recta, რაც მოგვიანებით, ბ.ვიჟინერის მიერ იყო გამოყენებული, როგორც დაშიფვრის ცენტრალური კომპონენტი.

ბ.ვიჟინერის მიერ 1586 წ. შემოთავაზებული მეთოდის ძირითადი არსი მსგავსია ცეზარის მეთოდისა. განსხვავება ამ ორ მეთოდს შორის მდგომარეობს შემდეგში: ცეზარის მეთოდში დაძვრის კოეფიციენტის მნიშვნელობა იყო მუდმივი სიდიდე, კერძოდ K=3; ხოლო ვიჟინერი თავის მეთოდში იყენებდა ინდივიდუალურ დაძვრის კოეფიციენტებს K_i_ს TI თითოეული S_i (i=1,2,…,Z) სიმბოლოს დასაშიფრავად.

ბ. ვიჟინერი, როგორც ცეზარი, მანიპულირებდა 26 სიმბოლოსაგან შემდგარ ლათინურ ალფავიტზე, რომლის მეშვეობით ის ადგენდა 26 სტრიქონიან ცხრილს (იხ. ცხრ. 4) რომელშიც ყოველი მომდევნო სტრიქონის სიმბოლოები იყო დაძრული მარცხნივ ერთი სიმბოლოთი. აღნიშნული წესით ფორმირებული ცხრილი იყო ექვივალენტური ცეზარის 26 ერთმანეთისაგან განსხვავებული ალფავიტისა (შიფრისა), რომელიც ცნობილია tabula recta-ს სახელით, ან აგრეთვე ვიჟინერის 26 სვეტისა და 26 სტრიქონის მქონე კვადრატის სახელწოდებით.

ცხრ. 4

დამშ TI სს K(K_i)	დასაშიფრი TI სტრიქონის სიმბოლოები (S_i)																									
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S

U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

სადაც, ცხრ. 4_ში „დამშ TI სს **K(K_i)“** - ნიშნავს „ დამიფრავი TI სტრიქონოს სიმბოლოები K(K_i)“

ძირითადი მოთხოვნა ბ. ვიჟინერი მიერ შემოთავაზებული მეთოდისა არის ის, რომ დასაშიფრი და დამშიფრავი TI სიგრძე უნდა იყოს ერთმანეთის ტოლი, ვინაიდან ყოველ S_i-იურ სიმბოლოს ინდივიდუალურად ეთანადება დაძვრის კოეფიციენტი K_i-იური. TI დაშიფვრის ამგვარი მიდგომა ქმნიდა ერთგვარ უხერხულობას იმ თვალსაზრისით, რომ როცა დასაშიფრი TI იყო დიდი მოცულობის, დამშიფრავი სტრიქონის ამავე რაოდენობის სიმბოლოების დამახსოვრება იწვევდა ერთგვარ უხერხულობას. აღნიშნულიდან თავის დაღწევის მიზნით, მიმართავდნენ შემდეგ ხერხს: დასაშიფრი TI-დან ამოიღებდნენ შედარებით მოკლე სიგრძის ფრაზას (დასამახსოვრებლად მოსახერხებელს) და ამ ფრაზის ციკლური გამეორების გზით აფორმირებდნენ დამშიფრავ სტრიქონს.

დაშიფვრის ალგორითმის არსი მდგომარეობს შემდეგში: ყოველ S_i და K_i სიმბოლოების წყვილს ჩაანაცვლებენ შიფროტექსტის (ShifTI) D_i სიმბოლოთი, რომელიც მდებარეობს ცხრ. 4_ის S_i-იურ სიმბოლოს შესაბამისი სვეტისა და K_i-იური სიმბოლოს შესაბამისი სტრიქონის გადაკვეთაზე. ანალოგიურად აღიწერება შიფროტექსტის გაშიფვრის ალგორითმი, კერძოდ: ყოველ D_i და K_i სიმბოლოების წყვილს ჩაანაცვლებენ S_i სიმბოლოთი, რომელიც განისაზღვრება ცხრ. 4_ის K_i-იურ სიმბოლოს სტრიქონში მდებარე D_i-იური სიმბოლოს შესაბამისი სვეტით.

განვიხილოთ კონკრეტული მაგალითი.

ვთქვათ, დასაშიფრია TI-ია „INFORMATIKA”, დახურული გასაღებით - K=„VALERI”. რადგან დასაშიფრ ტესტში შემავალი სიმბოლოების რაოდენობა აღემატება დამშიფრავ ტექსტში შემავალი სიმბოლოების რაოდენობას, ზემოაღნიშნული მოთხოვნების გათვალისწინებით, საჭიროა დამშიფრავ ტექსტს ციკლური გამეორების გზით დაემატოს საწყისი დამშიფრავი ტექსტი, ვიდრე არ გაუტოლდება დასაშიფრი და დამშიფრავი ტექსტების სიგრძეები. აღნიშნული პროცედურის შესრულების შედეგად მივიღებთ დამშიფრავ სტრიქონს (K) : VALERIVALER.

ტექსტის დაშიფრა ხორციელდება შემდეგი წესით (იხ.ცხრ.5): ShifTI_ის ყოველი D_i სიმბოლო განისაზღვრება ცხრ. 4_ში TI_ის S_i სვეტისა და K_i სტრიქონის შესაბამისი უჯრედების გადაკვეთაზე მდებარე უჯრედში შეტანილი სიმბოლოთი. ასე მაგალითად, თუ ავირჩევთ, რომ i=4, მაშინ $D_i \cong$ **T** მდებარეობს, როგორც ეს ცხრ.4_შია ნაჩვენები, მდებარეობს $S_i \cong$ **O** სვეტისა $K_i \cong$ **E** სტრიქონის შესაბამისი უჯრედების გადაკვეთაზე (სიმბოლო ≅ აღნიშნავს ექვივალენტობას).

ცხრ. 5

TI(S_i)	I	N	F	O	R	M	A	T	I	K	A
K(K_i)	V	A	L	E	R	I	V	A	L	E	R
ShifTI(D_i)	D	N	Q	S	A	U	V	T	T	O	R

დაშიფრული ტექსტყრი ინფორმაციის გაშიფრა ხორციელდება ანალოგიური წესით (ცხრ.6): TI_ის ყოველი S_i სიმბოლო განისაზღვრება ცხრ. 4_ში K_i სტრიქონში მდებარე D_i სიმბოლოს შესაბამისი სვეტის ამღნიშვნელი სიმბოლოთი. ასე მაგალითად, $K_i \cong$ **E** სიმბოლოს სვეტი განსაზღვრავს TI_ის S_i სიმბოლოს ანუ $S_i \cong$ **O.**

ცხრ. 6

ShifTI(D_i)	D	N	Q	S	A	U	V	T	T	O	R
K(K_i)	V	A	L	E	R	I	V	A	L	E	R
TI(S_i)	I	N	F	O	R	M	A	T	I	K	A

დაშიფვრისა და გაშიფვრის პროცედურების ვიზუალური რეალიზაცია ადვილად წარმოსადგენია და განსახორციელებელია დისკოების გამოყენებით, რომლებიც ნაჩვენებია ნახ. 2. აღნიშნული წესით ინფორმაციის დაშიფვრას და გაშიფვრას ძირითადად მიმართავდნენ ფრანგი სამხედროები სამოქალაქო ომების წარმოების დროს.

ვიჯინერის შიფრი დიდი ხნის განმავლობაში ითვლებოდა ძალზე მდგრად შიფრად. თუმცა ეს წარმოდგენა მასზე XIX საუკუნის დასაწყისში სრულიად გააქარწყლა კრიპტოანალიტიკოსმა კასისკიმ. ცნობილია აგრეთვე კრიპტოგრაფების მიერ ვიჯინერის შიფრის „გატეხვის“ შემთხვევები ჯერ კიდევ XVI საუკუნეში.

ნახ. 2. დაშიფვრა-გაშიფვრის ვიზუალური რეალიზაცია (დისკური ბარაბანი).

6.3 ვერნამის მეთოდი

გ.ვერნამმა, ამერიკის ტელეკომუნიკაციის და ტელეგრაფიის (AT & T) ფირმის თანამშრომელმა, 1917 წელს გამოიგონა, ხოლო 1919 წელს დააპატენტა ტელეგრაფიული შეტყობინებების ავტომატური დაშიფვრის სისტემა. მან შექმნა მოწყობილობა (აპარატი), რომელიც ახორციელებდა შეტყობინებების დაშიფვრას დამშიფრავის გარეშე. ამით ჩაეყარა საფუძველი ე.წ. „წრფივი დაშიფვრის“ მიმართულების განვითარებას, რომელიც ითვალისწინებდა შეტყობინებების დაშიფვრისა და მისი გადაცემის პროცედურების ავტომატურად და ერთდროულად შესრულებას. რაც თავის მხრივ საგრძნობლად ზრდიდა კავშირის ოპერატიულობას. გ. ვერნამის აპარატის ძირითად ნაწილს შეადგენდა კვანძი, ლოგიკური ფუნქციის შეკრება „ორის მოდულით“ მარეალიზებელი, რომელიც იყო აგებული რელეების ბაზაზე.

როგორც იყო აღნიშნული, გ.ვერნამის მიერ შემოთავაზებული კრიპტოსისტემა განკუთვნილი იყო ტელეგრაფური შეტყობინებების დასაშიფრავად, რომლებიც წარმოდგენილნი უნდა ყოფილიყო ბოდოს კოდში, ხუთნიშნა ორობით (ბინარულ) კოდში ცხრ. 7)

ცხრ.7

ბოდოს ორიგინალური კოდი	
მმართველი სიმბოლოები	
○. . . .	პრობელი, ასოების ცხრილზე გადასვლა
.○ . . .	პრობელი, ციფრების ცხრილზე გადასვლა
○○ . . .	ბოლო ნიშნის წაშლა

ასოების ცხრილი				**ციფრების ცხრილი**			
. . ○. .	A	○○ ○. .	K	. . ○. .	1	○. ○. .	.
. . ○○.	É	○○ ○○.	L	. . .○.	2	○. .○.	9/
. . .○.	E	○○ .○.	M	. . . .○	3	○. . .○	7/

..	.○○	I	○○	.○○	N	..	○.○	4	○. ○.○	2/
..	○○○	O	○○	○○○	P	..	○○○	5	○. ○○○	'
..	○.○	U	○○	○.○	Q	..	○○.	1/	○. ○○.	:
..	..○	Y	○○	..○	R	..	.○○	3/	○. .○○	?

.○	..○	B	○.	..○	S	.○	○..	6	○○ ○..	(
.○	○.○	C	○.	○.○	T	.○	.○.	7	○○ .○.	)
.○	○○○	D	○.	○○○	V	.○	..○	8	○○ ..○	-
.○	.○○	F	○.	.○○	W	.○	○.○	9	○○ ○.○	/
.○	.○.	G	○.	.○.	X	.○	○○○	0	○○ ○○○	+
.○	○○.	H	○.	○○.	Z	.○	○○.	4/	○○ ○○.	=
.○	○..	J	○.	○..	—	.○	.○○	5/	○○ .○○	£

ასოების და ციფრების ცხრილებში წრეწირები ტელეგრაფის ფირზე აღნიშნავენ ნახვრეტებს, რაც შეესაბამება S_i სიმბოლოს j-ური (j=1,2,3,4,5) ბიტის მნიშვნელობას - 1(ერთს), რომელიც წარმოდგენილია ხუთნიშნა ორობით (ბინარულ) კოდში. ანალოგიურად, . (წერტილი) შეესაბამება კოდში 0-ის მნიშვნელობას.

ამრიგად, შიფროტექსტის მისაღებად საკმარისია S_i (i=1,2,…,Z) სიმბოლოს ორობითი კოდის მნიშვნელობა შეიკრიბოს mod_2-ით წინასწარ შერჩეულ დახურული გასაღების ორობით კოდის მნიშვნელობასთან. ასე მაგალითად, თუ შერჩეული გასაღების ორობითი კოდი 10110-ია, ხოლო H სიმბოლოს კოდი 01110-ია, მაშინ დაშიფრულ D_i სიმბოლოს მიენიჭება მნიშვნელობა 10110 ⊕ 01110 = 11000 (სადაც, ⊕ -აღნიშნავს ლოგიკურ ოპერაციას შეკრებას mod_2-ით). თუ განმეორებით გავიმეორებთ იგივე პროცედურებს დაშიფრული სიმბოლოსა D_i და არჩეული გასაღების გამოყენებით მივიღებთ S_i-იურ სიმბოლოს კოდურ მნიშვნელობას (მაგ., 10110 ⊕ 11000 = 01110).

გ. ვერნამმა ჩამოაყალიბა დახურული გასაღების მიმართ სამი მოთხოვნა:

1. დახურული გასაღებების შერჩევა უნდა ხდებოდეს თანაბარი ალბათობით;
2. დახურული გასაღების სიგრძე უნდა იყოს ტოლი დასაშიფრი TI--ის სიგრძის;
3. დახურული გასაღები უნდა იქნეს გამოყენებული ერთხელ.

გ. ვერნამი მოითხოვდა აგრეთვე ფირის, რომელზეც იყო დაშიფრული შეტყობინება, განადგურებას მისი გამოყენების შემდეგ.

ცნობილია აგრეთვე გ. ,ვერნამის ე.წ. TI-ის mod_2-ით დაშიფვრის მეთოდი (მას 1918 წელს ეწოდა ვერნამ-ვიჯინერის შიფრი), რომელიც იყო რეზულტატი მისი მრავალგზის მცდელობისა გაეუმჯობინა ვიჯინერის მეთოდი. შემოთავაზებულ მეთოდში გ. ვერნამის ძირითადი მოთხოვნა იყო ის, რომ დახურული გასაღების სიგრძე აუცილებლად უნდა ყოფილიყო დასაშიფრი TI--ის სიგრძის ტოლი, რაც მეთოდის მომხმარებლების ერთგვარ უკმაყოფილებას იწვევდა (იხ. §2).

როგორც ცეზარი და ვიჟინერი, ვერნამიც მანიპულირებდა ლათინურ ალფავიტზე (A,B,C,…. ,Z). გ.ვერნამის დაშიფვრისა და გაშიფვრის პროცედურების რეალიზაცია შეიძლება აღიწეროს მოდულური არითმეტიკის ფორმულებით (იგულისხმება რომ სიმბოლოებს ალფავიტში მინიჭებული აქვთ რიგითი ნომრები დაწყებული ნულიდან, იხ. §1.1 ცხრ. 1), კერძოდ:

$$D_i = F((P_i^s+K_i)mod_n) \text{ და } S_i = f((P_i^D-K_i+n)mod_n) \quad (3)$$

$$D_i = F((P_i^S+K_i)mod_n) \text{ და } S_i = f((P_i^D-K_i+n)mod_n) \quad (4)$$

სადაც, P_i^S და P_i^D - S_i და D_i სიმბოლოების პოზიციებია (რიგითი ნომრებია) ალფავიტში, შესაბამისად, n - ალფავიტის სიმძლავრეა , K_i - გასაღები სიმბოლოს რიგითი ნომერია ალფავიტში, რომელიც მიუთითებს ალფავიტში S_i სიმბოლოდან მარჯვნივ K_i პოზიციით გადანაცვლების რიცხვითი მნიშვნელობაზე), F, f - ჩანაცვლების ოპერაციის მარეალიზებელი ლოგიკური ფუნქციებია.

განვიხილოთ კონკრეტული მაგალითი: სთქვათ უნდა დაიშიფროს TI –S=„GULNARA” დახურული გასაღებით K = „VALERII“. დაშიფვრისა და გაშიფვრის პროცედურული შეარულება ნაჩვენების ცხრ. 8 და ცხრ. 9, შესაბამისად.

ცხრ. 8

TI – S(S_i)	G - 6	U - 20	L - 11	N - 13	A - 0	R - 17	A - 0
K(K_i)	V - 21	A - 0	L - 11	E - 4	R - 17	I - 8	I - 8
ShifTI– (D_i)	B - 1	T - 20	W - 22	R - 17	R - 17	Z - 25	I - 8

ცხრ. 9

ShifTI– (D_i)	B - 1	T - 20	W - 22	R - 17	R - 17	Z - 25	I - 8
K(K_i)	V - 21	A - 0	L - 11	E - 4	R - 17	I - 8	I - 8
TI – S(S_i)	G - 6	U - 20	L - 11	N - 13	A - 0	R - 17	A - 0

შევნიშნოთ, რიცხვები ცხრილებში მიუთითებენ შესაბამისი სიმბოლოების რიგით ნომრებს (პოზიციებს) ალფაბიტში.

დასასრულს ავღნიშნოთ, რომ ვერნამის შიფრი დღემდე ითვლება პრაქტიკულად მდგრად შიფრად.

6.4 ტექსტური ინფორმაციის წარმოდგენის კოდური ფორმა

კრიპტოგრაფიაში ცნობილი მეთოდების მარეალიზებელი ალგორითმები ძირითადად დაფუძნებულია ერთი და იგივე პრინციპზე (იხ. §§1-3), რომლის ძირითადი არსი მდგომარეობს დასაშიფრ TI-ში შემავალი (შემცველი) სიმბოლოებზე ზოგიერთ წინასწარ განსაზღვრულ შეზღუდვების გათვალისწინებით, გარკვეული სახის მანიპულაციების ჩატარებაში. პირველ რიგში უნდა აღინიშნოს ის, რომ უმრავლესობა ცნობილი მეთოდების მარეალიზებელი ალგორითმები ორიენტირებულია ლათინური ალფავიტის გამოყენებაზე, მეორეს მხრივ გამოყენებულია ისეთი მარტივი მანიპულაციები (მაგალითად, ცეზარი, ვერნამი და ა.შ.), როგორიცაა სიმბოლოს რამოდენიმე პოზიციით დაძვრა მარჯვნივ ან მარცხნივ, სიმბოლოების ჩანაცვლება გარკვეული წესით და ა.შ.

თანამედროვე კომპიუტერული ტექნოლოგიების შესაძლებლობები იძლევა საშუალებას საგრძნობლად გაფართოვდეს კრიპტოგრაფიის ამოცანების განსაზღვრის არე, ძირითადად მოიხსნას ყოველგვარი შეზღუდვები. ასე მაგალითად, ქვემოთ აღწერილ მეთოდებში, დაშიფვრა/გაშიფვრის ალგორითმებში, გამოყენებულია არა მარტო ინგლისური ენის, არამედ ეროვნული (რუსული და ქართული) ენების ალფავიტების სიმბოლოების სრული ნაკრებები. გამოყენებულია აგრეთვე რიგი არითმეტიკული და ლოგიკური ოპერაციები ამ ნაკრებებზე სხვადასხვა სახის მანიპულაციების ჩასატარებლად.

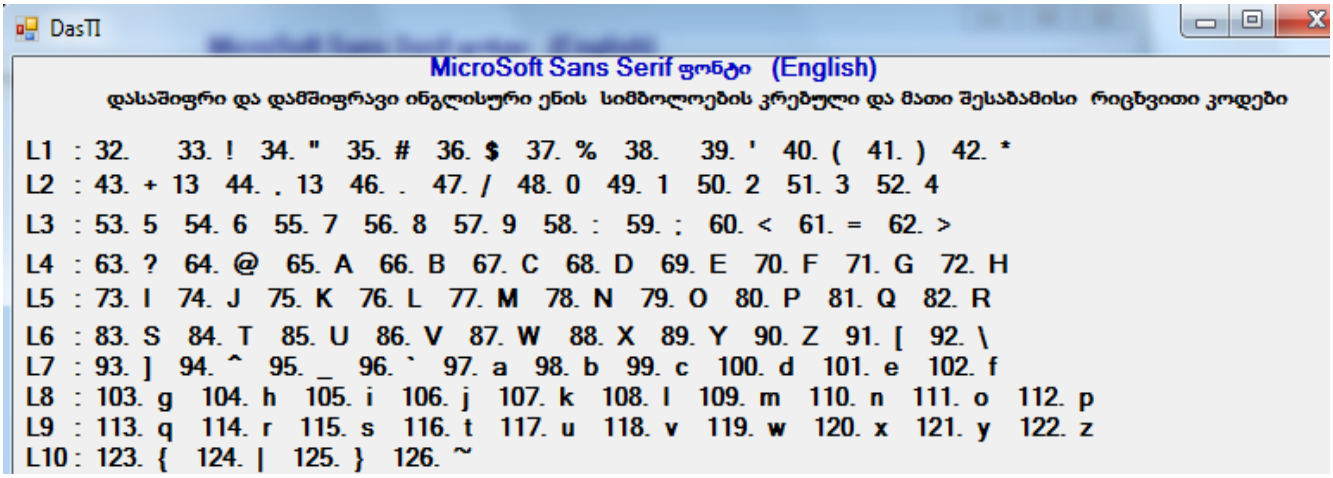
ცნობილია, რომ TI (რომელიც რა თქმა უნდა, სტრიქონული ტიპისაა) წარმოადგენს გამოყენებული ალფაბიტ(ებ)ის შემცველი სიმბოლოების S_i (სადაც $i=1,2,\ldots,Z$) გარკვეულ მიმდევრობას (ნაკრებს). ცნობილია აგრეთვე, რომ კომპიუტერში ყოველ სიმბოლოს ცალსახად შეესაბამება რიცხვითი მნიშვნელობა - S_i^k (კოდი), რომლებზედაც შეიძლება განხორციელდეს რიგი მანიპულაციებისა, კერძოდ, არითმეტიკული: შეკრება, გამოკლება და სხვა და ლოგიკური: შეკრება, გამრავლემა, შეკრება mod_n, ჩანაცვლება და სხვა, ოპერაციები. დასაშიფრ TI -ზე (DasTI) აღნიშნული არითმეტიკული და ლოგიკური ოპერაციების ჩატარების შედეგად მიიღება ე.წ. შიფროტექსტი- ShifTI (დაშიფრული ტექსტური ინფორმაცია), რომელიც ასევე განსაზღვრულია სიმბოლოების სიმრავლეზე, რომლისაც შეიძლება ეკუთვნოდეს DasTI სიმბოლოები.

ქვემოთ აღწერილ მეთოდებში, დაშიფვრისა და გაშიფვრის შემოთავაზებულ ალგორითმებში, გამოყენებულია ინგლისური (EN), რუსული (RU) და ქართული ენების (KA) Microsoft Sans Seris ფონტის შემცველი სიმბოლოების ნაკრებები და მათი შესაბამისი რიცხვითი კოდების მნიშვნელობები, რომლებიც ფიქსირდება კომპიუტერში პროგრამის Microsoft Visual studio 2010 (პრინციპით Default) ინსტალირების შედეგად. შევნიშნოთ, რომ შესაბამისი ენების {EN, RU და KA} გადამრთველი (ამომრჩევი) ღილაკები განთავსებულია Taskbar-ის ველში, მარჯვენა მხარეს.

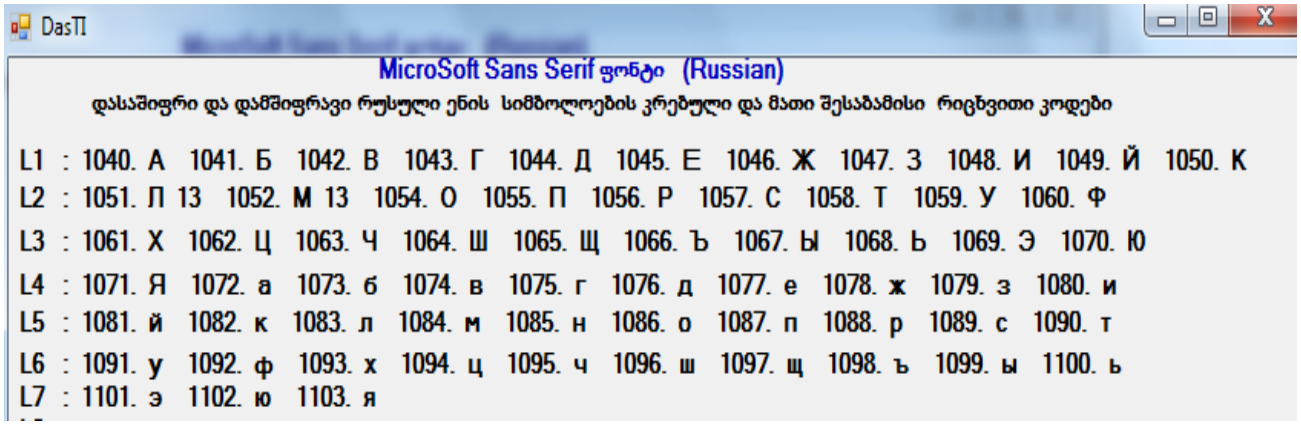
ცხრილებში 10-12 ნაჩვენებია, ზემოაღნიშნული ენების ფონტებში შემავალი სიმბოლოების ნაკრებები და აგრეთვე, ამ სიმბოლოების შესაბამისი მანქანური რიცხვითი კოდები, რომლებიც გამოიყენება ტექსტური ინფორმაციის DasTI და damTI დასამუშავებლად (დასაშიფვრად და

გასაშიფვრად). ავღნიშნოთ ეს ნაკრებები E_f, R_f , G_f, შესაბამისად, ხოლო αA-თი მათი ჯამური ნაკრებები E_f, R_f , G_f და ვუწოდოთ მას ალფავიტი -αA.

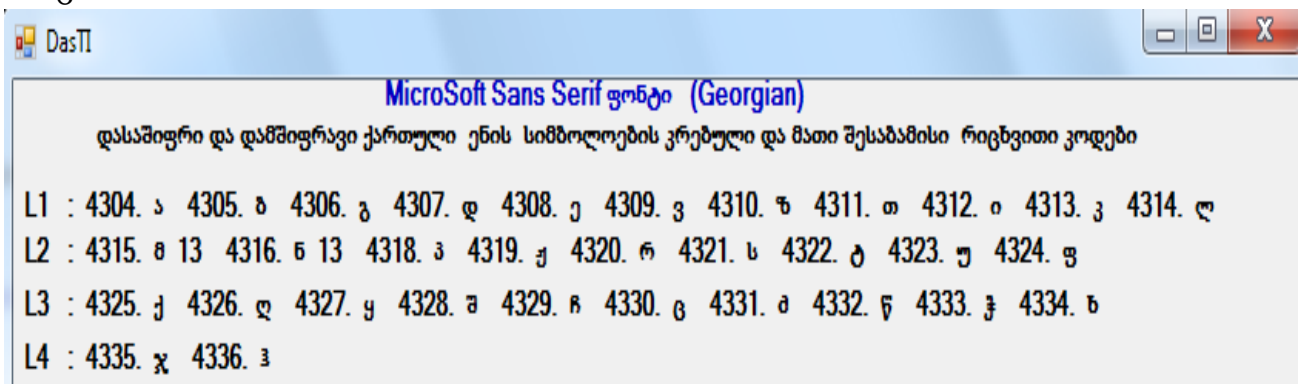
ცხრ. 10



ცხრ. 11



ცხრ. 12



ცხრილებში 13 -18 ნაჩვენებია, შიფროტექსტში - ShifTI შემავალი სიმბოლოების ნაკრებები და მათი შესაბამისი მანქანური რიცხვითი კოდების მნიშვნელობები. ავღნიშნოთ ეს ნაკრებები E_e, R_e , R_r, G_e , G_r , G_g შესაბამისად, ხოლო მათ ჯამურ ნაკრებების ვუწოდოთ ალფავიტი -βB.

ცხრ. 13

ShifTI

MicroSoft Sans Serif ფონტი (English)

შიფროტექსტის ინგლისური ენის სიმბოლოების კრებული და მათი შესაბამისი რიცხვითი კოდები

L1 :	64. @	65. A	66. B	67. C	68. D	69. E	70. F	71. G	72. H	73. I	74. J				
L2 :	75. K 13	76. L 13	78. N	79. O	80. P	81. Q	82. R	83. S	84. T						
L3 :	85. U	86. V	87. W	88. X	89. Y	90. Z	91. [	92. \	93.]	94. ^					
L4 :	95. _	96. `	97. a	98. b	99. c	100. d	101. e	102. f	103. g	104. h					
L5 :	105. i	106. j	107. k	108. l	109. m	110. n	111. o	112. p	113. q	114. r					
L6 :	115. s	116. t	117. u	118. v	119. w	120. x	121. y	122. z	123. {	124. |					
L7 :	125. }	126. ~	127. ▯	128.	129.	130.	131.	132.	133.	134.					
L8 :	135.	136.	137.	138.	139.	140.	141.	142.	143.	144.					
L9 :	145.	146.	147.	148.	149.	150.	151.	152.	153.	154.					
L10 :	155.	156.	157.	158.	159.	160.	161. ¡	162. ¢	163. £	164. ¤					
L11 :	165. ¥	166. ¦	167. §	168. ¨	169. ©	170. ª	171. «	172. ¬	173. -	174. ®					
L12 :	175. ¯	176. °	177. ±	178. ²	179. ³	180. ´	181. µ	182. ¶	183. ·	184. ¸					
L13 :	185. ¹	186. º	187. »	188. ¼	189. ½	190. ¾	191. ¿	192. À	193. Á	194. Â					
L14 :	195. Ã	196. Ä	197. Å	198. Æ	199. Ç	200. È	201. É	202. Ê	203. Ë	204. Ì					
L15 :	205. Í	206. Î	207. Ï	208. Ð	209. Ñ	210. Ò	211. Ó	212. Ô	213. Õ	214. Ö					
L16 :	215. ×	216. Ø	217. Ù	218. Ú	219. Û	220. Ü	221. Ý	222. Þ	223. ß	224. à					
L17 :	225. á	226. â	227. ã	228. ä	229. å	230. æ	231. ç	232. è	233. é	234. ê					
L18 :	235. ë	236. ì	237. í	238. î	239. ï	240. ð	241. ñ	242. ò	243. ó	244. ô					
L19 :	245. õ	246. ö	247. ÷	248. ø	249. ù	250. ú	251. û	252. ü	253. ý						

ცხრ. 14

ShifTI

MicroSoft Sans Serif ფონტი (Russian and English)

შიფროტექსტის ინგლისური და რუსული ენების სიმბოლოების კრებული და მათი შესაბამისი რიცხვითი კოდები

L1 :	1072. а	1073. б	1074. в	1075. г	1076. д	1077. е	1078. ж	1079. з	1080. и	1081. й	1082. к				
L2 :	1083. л 13	1084. м 13	1086. о	1087. п	1088. р	1089. с	1090. т	1091. у	1092. ф						
L3 :	1093. х	1094. ц	1095. ч	1096. ш	1097. щ	1098. ъ	1099. ы	1100. ь	1101. э	1102. ю					
L4 :	1103. я	1104. ѐ	1105. ё	1106. ђ	1107. ѓ	1108. є	1109. s	1110. і	1111. ї	1112. j					
L5 :	1113. ѣ	1114. ѣ	1115. ѣ	1116. ѣ	1117. ѣ	1118. ѣ	1119. ѣ	1120. ѣ	1121. ѣ	1122. ѣ					
L6 :	1123. ѣ	1124. ѣ	1125. ѣ	1126. ѣ	1127. ѣ	1128. ѣ	1129. ѣ	1130. ѣ	1131. ѣ	1132. ѣ					
L7 :	1133. ѣ	1134. ѣ	1135. ѣ	1136. ѣ	1137. ѣ	1138. ѣ	1139. ѣ	1140. ѣ	1141. ѣ	1142. ѣ					
L8 :	1143. ѣ	1144. ѣ	1145. ѣ	1146. ѣ	1147. ѣ	1148. ѣ	1149. ѣ	1150. ѣ	1151. ѣ	1152. ѣ					
L9 :	1153. ѣ	1154. ѣ	1155. ѣ	1156. ѣ	1157. ѣ	1158. ѣ	1159. ѣ	1160. ѣ	1161. ѣ	1162. ѣ					
L10 :	1163. ѣ	1164. ѣ	1165. ѣ	1166. ѣ	1167. ѣ	1168. ѣ	1169. ѣ	1170. ѣ	1171. ѣ	1172. ѣ					
L11 :	1173. ѣ	1174. ѣ	1175. ѣ	1176. ѣ	1177. ѣ	1178. ѣ	1179. ѣ	1180. ѣ	1181. ѣ	1182. ѣ					
L12 :	1183. ѣ	1184. ѣ	1185. ѣ	1186. ѣ	1187. ѣ	1188. ѣ	1189. ѣ	1190. ѣ	1191. ѣ	1192. ѣ					
L13 :	1193. ѣ	1194. ѣ	1195. ѣ	1196. ѣ	1197. ѣ	1198. ѣ	1199. ѣ	1200. ѣ	1201. ѣ	1202. ѣ					
L14 :	1203. ѣ	1204. ѣ	1205. ѣ	1206. ѣ	1207. ѣ	1208. ѣ	1209. ѣ	1210. ѣ	1211. ѣ	1212. ѣ					
L15 :	1213. ѣ	1214. ѣ	1215. ѣ	1216. ѣ	1217. ѣ	1218. ѣ	1219. ѣ	1220. ѣ	1221. ѣ	1222. ѣ					
L16 :	1223. ѣ	1224. ѣ	1225. ѣ	1226. ѣ	1227. ѣ	1228. ѣ	1229. ѣ	1230. ѣ	1231. ѣ	1232. ѣ					

ცხრ. 15

ShifTI

MicroSoft Sans Serif ფონტი (Russian and Russian)

შიფროტექსტის რუსული ენის სიმბოლოების კრებული და მათი შესაბამისი რიცხვითი კოდები

L1 :	2080. ▯	2081. ▯	2082. ▯	2083. ▯	2084. ▯	2085. ▯	2086. ▯	2087. ▯	2088. ▯	2089. ▯	2090. ▯
L2 :	2091. ▯	13 2092. ▯	13 2094. ▯	2095. ▯	2096. ▯	2097. ▯	2098. ▯	2099. ▯	2100. ▯		
L3 :	2101. ▯	2102. ▯	2103. ▯	2104. ▯	2105. ▯	2106. ▯	2107. ▯	2108. ▯	2109. ▯	2110. ▯	
L4 :	2111. ▯	2112. ▯	2113. ▯	2114. ▯	2115. ▯	2116. ▯	2117. ▯	2118. ▯	2119. ▯	2120. ▯	
L5 :	2121. ▯	2122. ▯	2123. ▯	2124. ▯	2125. ▯	2126. ▯	2127. ▯	2128. ▯	2129. ▯	2130. ▯	
L6 :	2131. ▯	2132. ▯	2133. ▯	2134. ▯	2135. ▯	2136. ▯	2137. ▯	2138. ▯	2139. ▯	2140. ▯	
L7 :	2141. ▯	2142. ▯	2143. ▯	2144. ▯	2145. ▯	2146. ▯	2147. ▯	2148. ▯	2149. ▯	2150. ▯	
L8 :	2151. ▯	2152. ▯	2153. ▯	2154. ▯	2155. ▯	2156. ▯	2157. ▯	2158. ▯	2159. ▯	2160. ▯	
L9 :	2161. ▯	2162. ▯	2163. ▯	2164. ▯	2165. ▯	2166. ▯	2167. ▯	2168. ▯	2169. ▯	2170. ▯	
L10 :	2171. ▯	2172. ▯	2173. ▯	2174. ▯	2175. ▯	2176. ▯	2177. ▯	2178. ▯	2179. ▯	2180. ▯	
L11 :	2181. ▯	2182. ▯	2183. ▯	2184. ▯	2185. ▯	2186. ▯	2187. ▯	2188. ▯	2189. ▯	2190. ▯	
L12 :	2191. ▯	2192. ▯	2193. ▯	2194. ▯	2195. ▯	2196. ▯	2197. ▯	2198. ▯	2199. ▯	2200. ▯	
L13 :	2201. ▯	2202. ▯	2203. ▯	2204. ▯	2205. ▯	2206. ▯	2207. ▯	2208. ▯	2209. ▯	2210. ▯	

ცხრ. 16

ShifTI

MicroSoft Sans Serif ფონტი (Georgian and English)

შიფროტექსტის ქართული და ინგლისური ენების სიმბოლოების კრებული და მათი შესაბამისი რიცხვითი კოდები

L1 :	4336. ᲗᲐ	4337. ᲘᲐ	4338. ᲘᲑ	4339. ᲘᲒ	4340. ᲘᲓ	4341. ᲘᲔ	4342. ᲘᲕ	4343. ᲘᲖ	4344. ᲘᲗ	4345. ᲘᲘ	4346. ᲘᲚ
L2 :	4347. ᲘᲛ	13 4348. ᲘᲜ	13 4350. ▯	4351. ▯	4352. ᲘᲞ	4353. ᲘᲟ	4354. ᲘᲠ	4355. ᲘᲡ	4356. ᲘᲢ		
L3 :	4357. ᲘᲣ	4358. ᲘᲤ	4359. ᲘᲥ	4360. ᲘᲦ	4361. ᲘᲧ	4362. ᲘᲨ	4363. ᲘᲩ	4364. ᲘᲪ	4365. ᲘᲫ	4366. ᲘᲬ	
L4 :	4367. ᲘᲭ	4368. ᲘᲮ	4369. ᲘᲯ	4370. ᲘᲰ	4371. ᲘᲱ	4372. ᲘᲲ	4373. ᲘᲳ	4374. ᲘᲴ	4375. ᲘᲵ	4376. ᲘᲶ	
L5 :	4377. ᲘᲷ	4378. ᲘᲸ	4379. ᲘᲹ	4380. ᲘᲺ	4381. Ი᲻	4382. Ი᲼	4383. ᲘᲽ	4384. ᲘᲾ	4385. ᲘᲿ	4386. Ი᳀	
L6 :	4387. Ი᳁	4388. Ი᳂	4389. Ი᳃	4390. Ი᳄	4391. Ი᳅	4392. Ი᳆	4393. Ი᳇	4394. Ი᳈	4395. Ი᳉	4396. Ი᳊	
L7 :	4397. Ი᳋	4398. Ი᳌	4399. Ი᳍	4400. Ი᳎	4401. Ი᳏	4402. Ი᳐	4403. Ი᳑	4404. Ი᳒	4405. Ი᳓	4406. Ი᳔	
L8 :	4407. Ი᳕	4408. Ი᳖	4409. Ი᳗	4410. Ი᳘	4411. Ი᳙	4412. Ი᳚	4413. Ი᳛	4414. Ი᳜	4415. Ი᳝	4416. Ი᳞	
L9 :	4417. Ი᳟	4418. Ი᳠	4419. Ი᳡	4420. Ი᳢	4421. Ი᳣	4422. Ი᳤	4423. Ი᳥	4424. Ი᳦	4425. Ი᳧	4426. Ი᳨	
L10 :	4427. Იᳩ	4428. Იᳪ	4429. Იᳫ	4430. Იᳬ	4431. Ი᳭	4432. Იᳮ	4433. Იᳯ	4434. Იᳰ	4435. Იᳱ	4436. Იᳲ	
L11 :	4437. Იᳳ	4438. Ი᳴	4439. Იᳵ	4440. Იᳶ	4441. Ი᳷	4442.	4443.	4444.	4445.	4446.	
L12 :	4447.	4448.	4449. ᅡ	4450. ᅢ	4451. ᅣ	4452. ᅤ	4453. ᅥ	4454. ᅦ	4455. ᅧ	4456. ᅨ	
L13 :	4457. ᅩ	4458. ᅪ	4459. ᅫ	4460. ᅬ	4461. ᅭ	4462. ᅮ	4463. ᅯ	4464. ᅰ	4465. ᅱ	4466. ᅲ	

ცხრ. 17

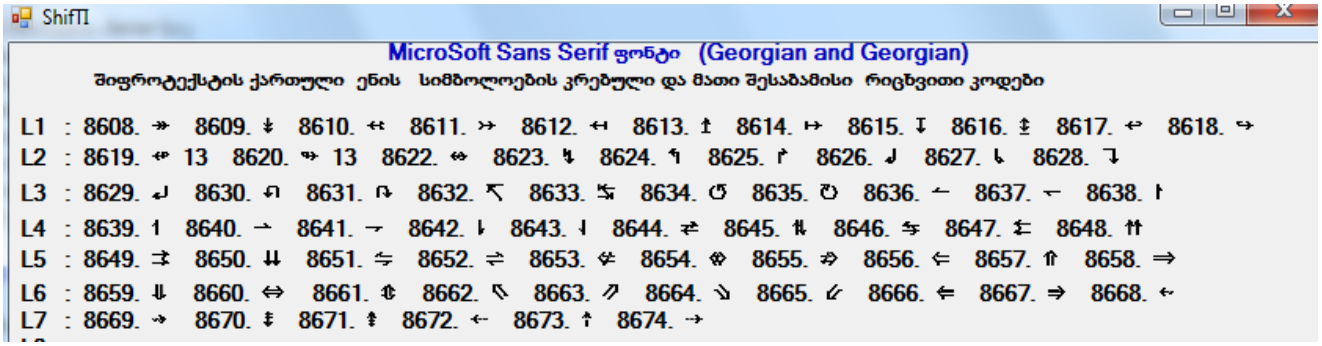
ShifTI

MicroSoft Sans Serif ფონტი (Georgian and Russian)

შიფროტექსტის ქართული და რუსული ენების სიმბოლოების კრებული და მათი შესაბამისი რიცხვითი კოდები

L1 :	5344. ᲐᲐ	5345. ᲐᲑ	5346. ᲐᲒ	5347. ᲐᲓ	5348. ᲐᲔ	5349. ᲐᲕ	5350. ᲐᲖ	5351. ᲐᲗ	5352. ᲐᲘ	5353. ᲐᲙ	5354. ᲐᲚ
L2 :	5355. ᲐᲛ	13 5356. ᲐᲜ	13 5358. ᲐᲞ	5359. ᲐᲟ	5360. ᲐᲠ	5361. ᲐᲡ	5362. ᲐᲢ	5363. ᲐᲣ	5364. ᲐᲤ		
L3 :	5365. ᲐᲥ	5366. ᲐᲦ	5367. ᲐᲧ	5368. ᲐᲨ	5369. ᲐᲩ	5370. ᲐᲪ	5371. ᲐᲫ	5372. ᲐᲬ	5373. ᲐᲭ	5374. ᲐᲮ	
L4 :	5375. ᲐᲯ	5376. ᲐᲰ	5377. ᲐᲱ	5378. ᲐᲲ	5379. ᲐᲳ	5380. ᲐᲴ	5381. ᲐᲵ	5382. ᲐᲶ	5383. ᲐᲷ	5384. ᲐᲸ	
L5 :	5385. ᲐᲹ	5386. ᲐᲺ	5387. Ა᲻	5388. Ა᲼	5389. ᲐᲽ	5390. ᲐᲾ	5391. ᲐᲿ	5392. ᲑᲐ	5393. ᲑᲑ	5394. ᲑᲒ	
L6 :	5395. ᲑᲓ	5396. ᲑᲔ	5397. ᲑᲕ	5398. ᲑᲖ	5399. ᲑᲗ	5400. ᲑᲘ	5401. ᲑᲙ	5402. ᲑᲚ	5403. ᲑᲛ	5404. ᲑᲜ	
L7 :	5405. ᲑᲝ	5406. ᲑᲞ	5407. ᲑᲟ	5408. ᲑᲠ	5409. ᲑᲡ	5410. ᲑᲢ	5411. ᲑᲣ	5412. ᲑᲤ	5413. ᲑᲥ	5414. ᲑᲦ	
L8 :	5415. ᲑᲧ	5416. ᲑᲨ	5417. ᲑᲩ	5418. ᲑᲪ	5419. ᲑᲫ	5420. ᲑᲬ	5421. ᲑᲭ	5422. ᲑᲮ	5423. ᲑᲯ	5424. ᲑᲰ	
L9 :	5425. ᲑᲱ	5426. ᲑᲲ	5427. ᲑᲳ	5428. ᲑᲴ	5429. ᲑᲵ	5430. ᲑᲶ	5431. ᲑᲷ	5432. ᲑᲸ	5433. ᲑᲹ	5434. ᲑᲺ	
L10 :	5435. Ბ᲻	5436. Ბ᲼	5437. ᲑᲽ	5438. ᲑᲾ	5439. ▪	5440. ᲑᲿ	5441. ᲒᲐ	5442. ᲒᲑ	5443. ᲒᲒ	5444. ᲒᲓ	

ცხრ. 18



ცხრ. 13 - ცხრ. 18 ჩანს სიმბოლოების კოდების მნიშვნელობების ცვლილების დიაპაზონი საგრძნობლად დიდია და მათი პრაქტიკული გამოყენება შეიძლება იყოს დაკავშირებული გარკვეულ სირთულეებთან. კერძოდ, შეუძლებელი ხდება დაშიფვრის ან/და გაშიფვრის ალგორითმების (PC გამოყენების გარეშე) სასწავლო მაგალითების ხელით შესრულება (კალკულატორის, თუნდაც საოფისე პროგრამის MS Excel შესაძლებლობების გამოყენებით). აღნიშნულის სირთულის თავიდან აცილების მიზნით შემოთავაზებულია ცხრ. 19 - ცხრ. 21 სიმბოლოების გაერთიანებული ნუსხა (იხ. ცხრ. 19-ის მე-3 სვეტი) და ამ ნუსხაში შემავალი ყოველი სიმბოლოს შესაბამისი კომპიუტერული კოდი (ცხრ. 19-ის მე-2 სვეტი), ხოლო ცხრ. 19-ის მე-1 სვეტში შეტანილია შემოთავაზებული კოდირება, ანუ კოდების რიცხვითი მნიშვნელობები, სიმბოლოების გაერთიანებული ნუსხისა..

ცხრ. 19 -ის მიხედვით სიმბოლოების ნაკრებთა რიცხვი 193-ის ტოლია (იხ. ცხრ. 19 და ცხრ. 20), რომელშიც სიმბოლოს მაქსიმალური კოდი არის 224, რაც იძლევა საშუალებას ავირჩიოთ ისეთი p მოდული(p მარტივი რიცხვია, გამოიყენება დაშიფვრის დახურული გასაღების გამოსათვლელად), რომელიც უნდა აკმაყოფილებდეს პირობას p >224 და არა პირობას p >4336. სიმბოლოების აღწერილი ნაკრების და მათი შესაბამისი კოდების გამოყენებით (იხ. ცხრ. 20 ანუ ე.წ. - NαA ალფაბიტი და ცხრ. 21 ანუ ე.წ. - NβB ალფავიტი) დაპროგრამებულია ზოგიერთი მეთოდები. კერძოდ სიმეტრიული სისტემის უნივერსალური მეთოდი, ხოლო ასიმეტრიული სისტემის რაივესტ- შამირ- ეიდელმანის მეთოდი .

დასასრულს შევნიშნოთ, შემოთავაზებული ყველა პრროგრამული მოდული (რომელსაც მინიჭებული აქვს მასში რეალიზებული მეთოდის სახელი) განთავსებულია D-ხისტ დისკოზე საქაღალდეში D:\Kripto_VaKe_GuKo. შევნიშნოთ, აგრეთვე,რომ დაპროგრამებულ მეთოდების რიცხვს მიეკუთვნებიან სიმეტრიული სისტემების : ცეზარის, ვიჟინერის, ვერნამის, უნივერსალური და შებრუნებული მატრიცის მეთოდები.

ცხრ. 19. კომპიუტერული სიმბოლოების ნაკრები

№	1	2	3
1	32	32	ჰაერი
2	33	33	!
3	34	34	"
4	35	35	#
5	36	36	$
6	37	37	%
7	38	38	&
8	39	39	აპოსროფი
9	40	40	(
10	41	41	)
11	42	42	*
12	43	43	+
13	44	44	.
14	45	45	-
15	46	46	,
16	47	47	/
17	48	48	0
18	49	49	1
19	50	50	2
20	51	51	3
21	52	52	4
22	53	53	5
23	54	54	6
24	55	55	7
25	56	56	8
26	57	57	9
27	58	58	:
28	59	59	;
29	60	60	<
30	61	61	=
31	62	62	>
32	63	63	?
33	64	64	@
34	65	65	A
35	66	66	B
36	67	67	C
37	68	68	D
38	69	69	E
39	70	70	F
40	71	71	G
41	72	72	H
42	73	73	I
43	74	74	J

№	1	2	3
44	75	75	K
45	76	76	L
46	77	77	M
47	78	78	N
48	79	79	O
49	80	80	P
50	81	81	Q
51	82	82	R
52	83	83	S
53	84	84	T
54	85	85	U
55	86	86	V
56	87	87	W
57	88	88	X
58	89	89	Y
59	90	90	Z
60	91	91	[
61	92	92	\
62	93	93	]
63	94	94	^
64	95	95	_
65	96	96	`
66	97	97	a
67	98	98	b
68	99	99	c
69	100	100	d
70	101	101	e
71	102	102	f
72	103	103	g
73	104	104	h
74	105	105	i
75	106	106	j
76	107	107	k
77	108	108	l
78	109	109	m
79	110	110	n
80	111	111	o
81	112	112	p
82	113	113	q
83	114	114	r
84	115	115	s
85	116	116	t
86	117	117	u

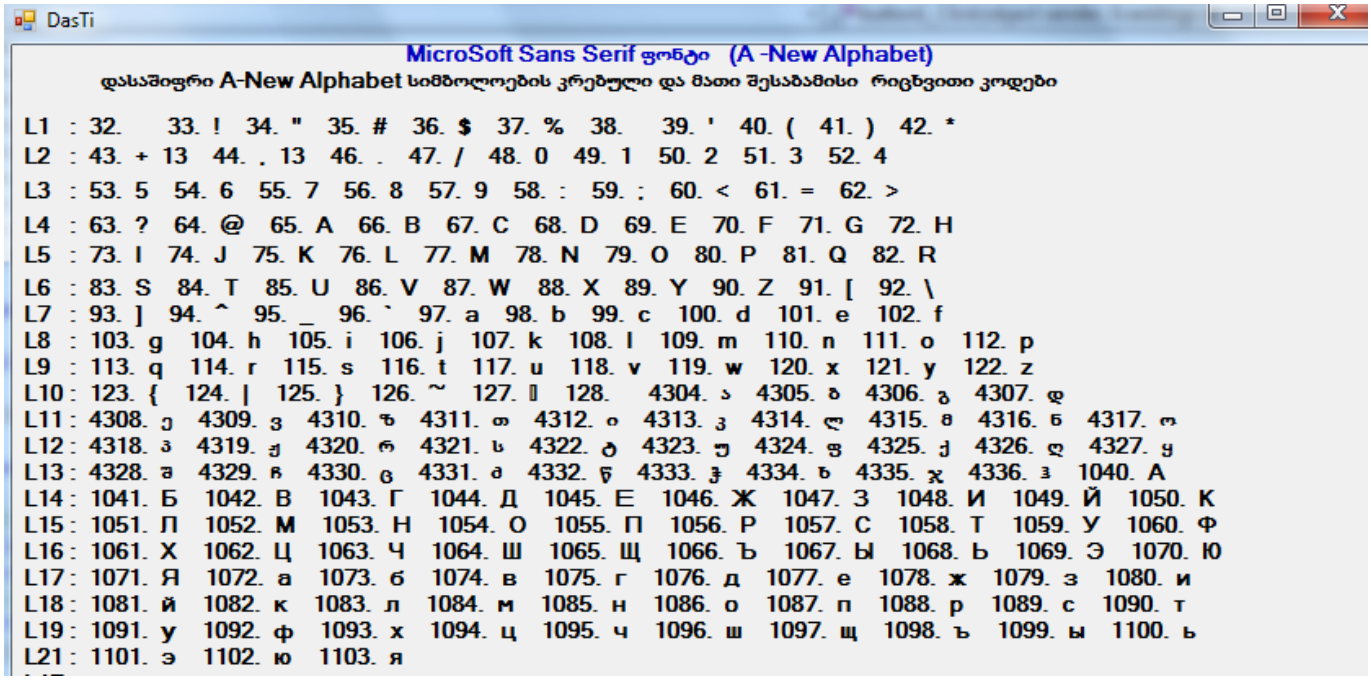
№	1	2	3
87	118	118	v
88	119	119	w
89	120	120	x
90	121	121	y
91	122	122	z
92	123	123	{
93	124	124	|
94	125	125	}
95	126	126	~
96	127	127	
97	128	4304	ა
98	129	4305	ბ
99	130	4306	გ
100	131	4307	დ
101	132	4308	ე
102	133	4309	ვ
103	134	4310	ზ
104	135	4311	თ
105	136	4312	ი
106	137	4313	კ
107	138	4314	ლ
108	139	4315	მ
109	140	4316	ნ
110	141	4317	ო
111	142	4318	პ
112	143	4319	ჟ
113	144	4320	რ
114	145	4321	ს
115	146	4322	ტ
116	147	4323	უ
117	148	4324	ფ
118	149	4325	ქ
119	150	4326	ღ
120	151	4327	ყ
121	152	4328	შ
122	153	4329	ჩ
123	154	4330	ც
124	155	4331	ძ
125	156	4332	წ
126	157	4333	ჭ
127	158	4334	ხ
128	159	4335	ჯ
129	160	4336	ჰ

№	1	2	3
130	161	1040	А
131	162	1041	Б
132	163	1042	В
133	164	1043	Г
134	165	1044	Д
135	166	1045	Е
136	167	1046	Ж
137	168	1047	З
138	169	1048	И
139	170	1049	Й
140	171	1050	К
141	172	1051	Л
142	173	1052	М
143	174	1053	Н
144	175	1054	О
145	176	1055	П
146	177	1056	Р
147	178	1057	С
148	179	1058	Т
149	180	1059	У
150	181	1060	Ф
151	182	1061	Х
152	183	1062	Ц
153	184	1063	Ч
154	185	1064	Ш
155	186	1065	Щ
156	187	1066	Ъ
157	188	1067	Ы
158	189	1068	Ь
159	190	1069	Э
160	191	1070	Ю
161	192	1071	Я
162	193	1072	а
163	194	1073	б
164	195	1074	в
165	196	1075	г
166	197	1076	д
167	198	1077	е
168	199	1078	ж
169	200	1079	з
170	201	1080	и
171	202	1081	й
172	203	1082	к

№	1	2	3
173	204	1083	л
174	205	1084	м
175	206	1085	н
176	207	1086	о
177	208	1087	п
178	209	1088	р
179	210	1089	с
180	211	1090	т
181	212	1091	у
182	213	1092	ф
183	214	1093	Х

№	1	2	3
184	215	1094	ц
185	216	1095	Ч
186	217	1096	ш
187	218	1097	щ
188	219	1198	ъ
189	220	1199	ы
190	221	1100	ь
191	222	1101	э
192	223	1102	ю
193	224	1103	я

ცხრ. 20



ცხრ. 21

ShifTI

MicroSoft Sans Serif ფონტი (A -New Alphabet)

შიფროტექსტის A-New Alphabet სიმბოლოების კრებული და მათი შესაბამისი რიცხვითი კოდები

L1 : 64. @ 65. A 66. B 67. C 68. D 69. E 70. F 71. G 72. H 73. I 74. J
L2 : 75. K 13 76. L 13 78. N 79. O 80. P 81. Q 82. R 83. S 84. T
L3 : 85. U 86. V 87. W 88. X 89. Y 90. Z 91. [92. \ 93.] 94. ^
L4 : 95. _ 96. ` 97. a 98. b 99. c 100. d 101. e 102. f 103. g 104. h
L5 : 105. i 106. j 107. k 108. l 109. m 110. n 111. o 112. p 113. q 114. r
L6 : 115. s 116. t 117. u 118. v 119. w 120. x 121. y 122. z 123. { 124. |
L7 : 125. } 126. ~ 127. ▯ 128. 129. 130. 131. 132. 133. 134.
L8 : 135. 136. 137. 138. 139. 140. 141. 142. 143. 144.
L9 : 145. 146. 147. 148. 149. 150. 151. 152. 153. 154.
L10 : 155. 156. 157. 158. 159. 160. 161. ¡ 162. ¢ 163. £ 164. ¤
L11 : 165. ¥ 166. ¦ 167. § 168. ¨ 169. © 170. ª 171. « 172. ¬ 173. - 174. ®
L12 : 175. ¯ 176. ° 177. ± 178. ² 179. ³ 180. ´ 181. µ 182. ¶ 183. · 184. ¸
L13 : 185. ¹ 186. º 187. » 188. ¼ 189. ½ 190. ¾ 191. ¿ 192. À 193. Á 194. Â
L14 : 195. Ã 196. Ä 197. Å 198. Æ 199. Ç 200. È 201. É 202. Ê 203. Ë 204. Ì
L15 : 205. Í 206. Î 207. Ï 208. Ð 209. Ñ 210. Ò 211. Ó 212. Ô 213. Õ 214. Ö
L16 : 215. × 216. Ø 217. Ù 218. Ú 219. Û 220. Ü 221. Ý 222. Þ 223. ß 224. à
L17 : 225. á 226. â 227. ã 228. ä 229. å 230. æ 231. ç 232. è 233. é 234. ê
L18 : 235. ë 236. ì 237. í 238. î 239. ï 240. ð 241. ñ 242. ò 243. ó 244. ô
L19 : 245. õ 246. ö 247. ÷ 248. ø 249. ù 250. ú 251. û 252. ü 253. ý 254. þ
L21 : 255. ÿ 256. Ā 257. ā 258. Ă 259. ă 260. Ą 261. ą 262. Ć 263. ć 264. Ĉ

ცხრ.22

L1 : 265. ĉ 266. Ċ 267. ċ 268. Č 269. č 270. Ď 271. ď 272. Đ 273. đ 274. Ē 275. ē
L2 : 276. Ĕ 13 277. ĕ 13 279. ė 280. Ę 281. ę 282. Ě 283. ě 284. Ĝ 285. ĝ
L3 : 286. Ğ 287. ğ 288. Ġ 289. ġ 290. Ģ 291. ģ 292. Ĥ 293. ĥ 294. Ħ 295. ħ
L4 : 296. Ĩ 297. ĩ 298. Ī 299. ī 300. Ĭ 301. ĭ 302. Į 303. į 304. İ 305. ı
L5 : 306. IJ 307. ij 308. Ĵ 309. ĵ 310. Ķ 311. ķ 312. ĸ 313. Ĺ 314. ĺ 315. Ļ
L6 : 316. ļ 317. Ľ 318. ľ 319. Ŀ 320. ŀ 321. Ł 322. ł 323. Ń 324. ń 325. Ņ
L7 : 326. ņ 327. Ň 328. ň 329. ʼn 330. Ŋ 331. ŋ 332. Ō 333. ō 334. Ŏ 335. ŏ
L8 : 336. Ő 337. ő 338. Œ 339. œ 340. Ŕ 341. ŕ 342. Ŗ 343. ŗ 344. Ř 345. ř
L9 : 346. Ś 347. ś 348. Ŝ 349. ŝ 350. Ş 351. ş 352. Š 353. š 354. Ţ 355. ţ
L10 : 356. Ť 357. ť 358. Ŧ 359. ŧ 360. Ũ 361. ũ 362. Ū 363. ū 364. Ŭ 365. ŭ
L11 : 366. Ů 367. ů 368. Ű 369. ű 370. Ų 371. ų 372. Ŵ 373. ŵ 374. Ŷ 375. ŷ
L12 : 376. Ÿ 377. Ź 378. ź 379. Ż 380. ż 381. Ž 382. ž 383. ſ 384. ƀ 385. Ɓ
L13 : 386. Ƃ 387. ƃ 388. Ƅ 389. ƅ 390. Ɔ

6.5 სიმეტრიული სისტემის მეთოდების უნივერსალური მოდელი

როგორც იყო აღნიშნული, დასაშიფრი ტექსტური ინფორმაცია DasTI - ეს არის სიმბო ლოების მიმდევრობა - DasTI={ S_i} (სადაც , i=1,2,3, ...,Z; Z – სიმბოლოების მაქსიმალური რიცხვია DasTI-ში), რომლებიც პკ შეიტანება კლავიატურიდან. ყოველი სიმბოლო S_i ეკუთვნის ამორჩეულ ალფავიტს, რომელსაც თავის მხრივ, ცალსახად შეესაბამება გარკვე ული რიცხვითი მნიშვნელობა - მისი შესაბამისი კოდი (იხ. ცხრ. 10-21). ამგვარად, DasTI შეიძლება აგრეთვე იქნას წარმოდგენილი დადებით მთელ რიცხვთა მიმდევრობით : DasTIK={S_i^k} სადაც,

-DasTIK - არის DasTI_ში შემავალი სიმბოლოების კოდური რიცხვითი მნიშვნელობები ,

-S_i^k - არის S_i სიმბოლოს რიცხვითი კოდი (i=1,2,3, ...,Z).

DasTIK--ები წარმოადგენენ დაშიფვრისა და გაშიფვრის ობიექტებს, მათზე ხორციელდება გარკვეული გარდასახვები, მანიპულაციები (არითმეტიკული და ლოგიკური ოპერაციები), რის შედეგადაც მიიღება რიცხვების ახალი მიმდევრობა - ShiTIK (შიფროტექსტის სიმბოლოების კოდები: ShiTIK= {D_i^k} , i=1,2,3, ...,Z;); ნებისმიერი D_i^k შეიძლება ზოგად შემთხვევაში იყოს განსაზღვრული სასრულო რიცხვთა სიმრავლეზე , რომლის თითოეულ ელემენტს ცალსახად შეესაბამება გარკვეული სიმბოლო (კერძოდ, Microsoft Sans Seris ფონტის) ფონტების იმ კრებულიდან, რომელებიც გენერირებულია პკ (იხ. ცხრ. 13 -18). DasTIK-ზე მანიპულაციების ჩატარების მიზნით განიხილავენ აგრეთვე დამშიფრავი სიმბოლოების DamTI ={ K_j} შესაბამის კოდურ მნიშვნელობებს DamTIK={K_j^k} , სადაც , (j=1,2,3, ...,z<=Z),

-DamTI -დამშიფრავი ტექსტური ინფორმაციაა,

-DamTIK – DamTI შემავალი სიმბოლოების შესაბამისი კოდებია , რომლებიც ეკუთვნიან αA ან NαA ალფაბიტს. იმისა და მიხედვით თუ რას უდრის j ცვლადის მნიშვნელობა , განიხილავენ სიმეტრიული სისტემის ამა თუ იმ მეთოდს. კერძოდ , თუ:

ა) j =1 , აღნიშნავენ რომ TI დაშიფვრა / გაშიფვრა ხორციელდება ცეზარის მეთოდით,

ბ) 1<j< Z, აღნიშნავენ რომ TI დაშიფვრა / გაშიფვრა ხორციელდება ვიჟინერის მეთოდით,

გ) j= Z, აღნიშნავენ რომ TI დაშიფვრა / გაშიფვრა ხორციელდება ვერნამის მეთოდით.

TI დაშიფვრისა -გაშიფვრის პროცედურის არსი მდგომარეობს შემდეგში: ყოველ S_i^K მნიშვნელობას DasTIK მიმდევრობიდან უთანადებენ DamTIK მიმდევრობიდან K_{j+1}^k მნიშვნელობას და აწარმოებენ მათზე წინასწარ განსაზღვრულ მანიპულაციებს, სადაც

$$j=(i-1)\%z \qquad (5)$$

z<= Z - ამორჩეული სიმბოლოების საწყისი რაოდენობაა დამშიფრავ სტრიქონში.

(5) ფორმულიდან ჩანს, რომ დამშიფრავი სიმბოლოს ან სიმბოლოების (z) ჯგუფის მონაწილეობა დაშიფვრა - გაშიფვრის პროცესში არ აღემატება Z-ს. აქედან გამომდინარე, დამშიფრავი სტრიქონის ეფექტურად ფორმირების მიზნით განვიხილოთ შეფარდება

$$g=Z/z$$

სადაც , g მთელი დადებითი რიცხვია აღებული მეტობით და გვიჩვენებს დამშიფრავ სტრიქონში ამორჩეული საწყისი სიმბოლოების გამეორების ჯერადობას. ცხადია

$$Z<=g*z \qquad (6)$$

იმ შემთხვევაში თუ აღმოჩნდება, რომ ფორმულაში (6) ადგილი აქვს უტოლობას, ბოლო (g*z - Z) სიმბოლო არ მიიღებს მონაწილეობას დაშიფვრა - გაშიფრის პროცედურების რეალიზაციაში, რაც ნიშნავს იმას, რომ DamTI სტრიქონი შეიძლება წარმოვდგინოთ DasTI სტრიქონის ანალოგიურად ანუ DamTI={ K_i} ანდა DamTIK-{ K_i^k } სახით. აღნიშნულიდან გამომდინარე, ადგილი აქვს დაშიფვრა / გაშიფვრის პროცედურის ფორმლური სახით:

$$F^k(S_i^K, K_i^k) = D_i^K \qquad F(S_i, K_i) = D_i \qquad (7)$$

$$f^k(D_i^K, K_i^k) = S_i^K \qquad f(D_i, K_i) = S_i \qquad (8)$$

გამოსახულებების (7) და (8) თანახმად :

ა) TI დასაშიფვრად ყოველთვის მოიძებნება ერთი მაინც სიმრავლე - αA/ NαA (S_i^k, $D_i^k \in$ αA/ NαA) და ფუნქცია F^k (F), რომლის შესრულების შედეგი - D_i^K (D_i) იქნება განსაზღვრული βB/ NαB - სასრულო სიმრავლეზე (შევნიშნოთ, რომ αA/ NαA და βB/ NβB გადაკვეთა შეიძლება არ იყოს ცარიელი ; იხ. ცხრ. 10 -21).

ბ) თუ არსებობს βB/ NβB სიმრავლე, სადაც D_i^k Є βB/ NβB , K_i^K Є βB/ NβB ყოველთვის მოიძებნება F^k (F) ფუნქციის შებრუნებული ფუნქცია f^k (f) რომლის შესრულების შედეგი - S_i^K (S_i) იქნება განსაზღვრული - αA/ NαA სასრულო სიმრავლეზე (იხ. ცხრ. 1 და 12).

ავღნიშნოთ, რომ ცნობილ სიმეტრიულ სისტემებში F^k ფუნქციის ქვეშ , როგორც წესი , იგულისხმება არითმეტიკული ოპერაციები (კერძოდ, ოპერაცია „შეკრება“) ან/ და ლოგიკური ოპერაცია „შეკრება mod_2“, ხოლო f^k ფუნქციის ქვეშ კი არითმეტიკული ოპერაციები (კერძოდ , ოპერაცია „გამოკლება“) ან/და ლოგიკური ოპერაცია „შეკრება mod_2“.

უნივერსალური მეთოდის რეალიზაცია

განვიხილოთ კონკრეტულ მაგალითზე უნივერსალური მეთოდით DasTI დაშიფვრა - გაშიფვრის პროცედურურული შესრულების თანმიმდევრობა , რომელიც ნაჩვენებია ცხრ. 13-15 (იხ. აგრეთვე ნახ. 3 და 4). იგულისხმება, რომ DasTI-ის (ვთქვათ, “VaლერიЙ”) და damTI-ის (ვთქვათ, “Aბ” , აქ z=2, g=4) ფორმირება ხორციელდება სიმბოლოების ნაკრებით, რომელიც მოცემულია ცხრ. 10-ის მესამე სვეტში (იხ. ცხრ. 11). ამ ნაკრებებით ცხრ. 13 (ცხრ.14) პირველი სტრიქონის 2:8 (2-3) სვეტებში შეტანილია DasTI (DamTI) შემადგენელი სიმბოლოები , ხოლო სტრიქონ 2-ში მათი შესაბამისი კოდური მნიშვნელობები (მანქანური კოდები), აღებული ცხრ. 10-ის მეორე სვეტიდან - S_i^k(ს2) (D_i^k(ს2)); ცხრ. 13-ის მე-3 სტრიქონში შეტანილია (1) და (2) ფორმულებით განსაზღვრული წესით ფორმირებული damTI, ხოლო მისი სიმბოლოების მანქანური კოდები (D_i^k(ს2)), აღებული ცხრ. 10-ის მეორე სვეტიდან, შეტანილია მე-4 სტრიქონში. ცხრ. 13-ის მე-5 და მე-6 სტრიქონებში შესაბამისად შეტანილია DasTI და DamTI სიმბოლოების რიცხვითი კოდების მნიშვნელობები {(S_i^k(ს1) და D_i^k(ს1)} აღებული ცხრ. 10-ის პირველი სვეტიდან (იხ. აგრეთვე ცხრ. 14 მე- 3 სტრიქონი). მე-7 სტრიქონსი დაფიქსირებულია F^k ფუნქციის (მარეალიზებელი ოპერაცია „შეკრება“) შესრულების შედეგი (იხ. ფორმულა (3)). მე-8 სტრიქონში ნაჩვენებია შიფროტექსტი, რომლის სიმბოლოები განსაზღვრულია მე-7 სტრიქონში ასახული მანქანური კოდებით ცხრ. 12-დან (ჩაწერილია კოდების შესაბამისი სიმბოლოების მიახლოებითი მსგავსებით).

როგორც ცნობილია, შიფროტექსტი (F^k - Q_i^k შედეგი მე-7 სტრიქონში მოცემული სახით) და DamTI (გამშიფრავი გასაღები) გადაეცემა TI გამშიფრავს. შევნიშნოთ, რომ სიმეტრიულ სისტემებში TI დასაშიფრავად და გასაშიფრავად გამოიყენება ერთი და იგივე ტექსტური ინფორმაცია. ShiTI –ის გაშიფვრის პროცედურული შესრულების თანმიმდევრობა ნაჩვენებია ნახ. 4 და ცხრ. 15.

ცხრ. 13

№	1	2	3	4	5	6	7	8
1	DasTI	V	a	ლ	ე	р	и	Й
2	S_i^k(ს2)	86	97	4314	4308	1088	1080	1049
3	DamTI	A	ბ	A	ბ	A	ბ	A
4	D_i^k(ს2)	65	4305	65	4305	65	4305	65
5	S_i^k(ს1)	86	97	138	132	209	201	170
6	D_i^k(ს1)	65	129	65	129	65	129	65
7	F^k - Q_i^k (ცხ. 12)	151	226	203	261	274	330	235
8	ShiTI		^a	~E	a|	~E	N,	`e

DamTI სიმბოლოები და მათი კოდების მნიშვნელობები

ცხრ. 14

№	1	2	3
1	DamTI	A	ბ
2	D_i^k(ს2)	65	4305
3	D_i^k(ს1)	65	129

მე - 15 ცხრილის:

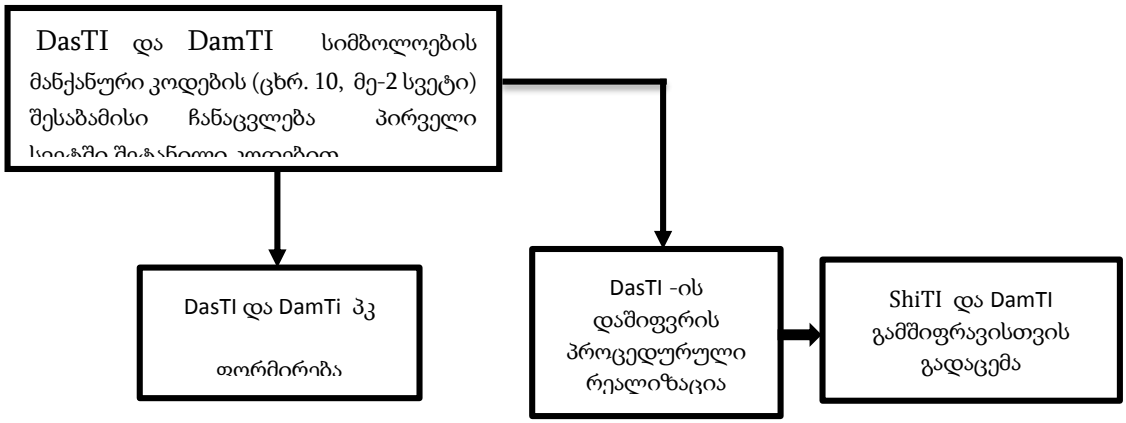
- მეცხრე სტრიქონში ნაჩვენებოია შიფროტექსტი, რომელიც DamTI ერტად გედაეგზავნება გამშიფრავს;
- მეათე სტრიქონში შეტანილია დაშიფრული ტექსტში შემავალი სიმბოლოების კოდების მნიშვნელობები (იხ. ცხრ.14, მე-7 სტრიქონი);
- მეტერთმეტე სტრიქონში შეტანილია გამშიფრავი ტექსტში DamTI შემავალი სიმბოლოები (იხ. ცხრ.14, მე-3 სტრიქონი);
- მე-12 და მე - 13 სტრიქონების ფორმირება წარმოებს ანალოგიურად , როგორც ეს იყო შესრულებული მე-4 და მე- 6 სტიქონების ფორმირებისას;
- მე-14 სტრიქონში შეტანილია გაშიფვრის ფუნქციის - f^k , ოპერაცია „გამოკლება“, შედეგი;

- მე-15 სტრიქონი შევსებულია კოდების მნიშვნელობებით , რომლებიც მიიღება მათი შესაბამისი ჩანაცვლებით ცხრ. 10-ის მე-2 სვეტის მნიშვნელობებით;

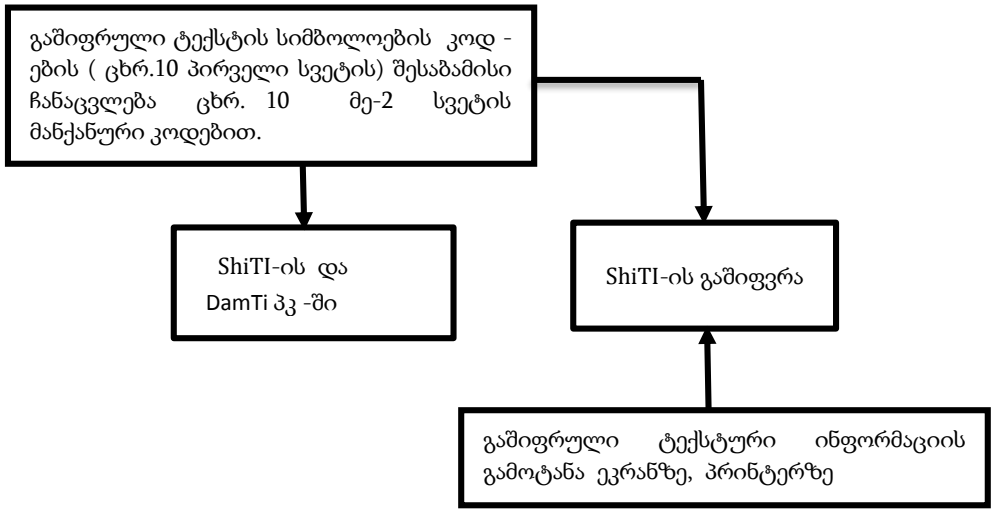
- მე-16 სტრიქონში ასახულია DasTI, რომელშიც შემავალი სიმბოლოების კოდების მნიშვნელობები სტრიქონ 15 - შია ჩაწერილი (იხ. ცხრ. 10).

ცხრ. 15

№	1	2	3	4	5	6	7	8
1.	ShiTI		^a	~E	a|	~E	N,	`e
2.	F^k-Q_i^k (ცხ.12)	151	226	203	261	274	330	235
3.	DamTI	A	ბ	A	ბ	A	ბ	A
4.	D_i^k(ს2)	65	4305	65	4305	65	4305	65
5.	D_i^k(ს1)	65	129	65	129	65	129	65
6.	f^k S_i^k(ს1)	86	97	138	132	209	201	170
7.	S_i^k(ს2)	86	97	4314	4308	1088	1080	1049
8.	DasTI	V	a	ლ	ე	р	и	Й



ნახ. 3. საწყისი ტექსტური ინფორმაციის დაშიფვრის პროცედურა.



ნახ. 4. დაშიფრული ტექსტური ინფორმაციის გაშიფვრის პროცედურა.

6.6. სიმეტრიული სისტემების უნივერსალური მოდელის მიკროპროგრამული სახით წარმოდგენა და მისი სქემო-ტექნიკური რეალიზაცია

ცეზარის, ვიჯინერისა და ვერნამის მეთოდების ანალიზმა გვიჩვენა, რომ ისინი ძირითადში ერთგვაროვანნი არიან პრაქტიკული რეალიზაციის თვალსაზრისით. ეს აადვილებს ამ მეთოდების ერთიან ასპექტში აბსტრაქტული მოდელის სახით წარმოდგენას და მის არა მარტო პროგრამულ რეალიზაციას, არამედ სქემო - ტექნიკურ რეალიზაციასაც. აღნიშნული მეთოდების მარეალიზებელი ალგორითმების ფორმალური ენით აღწერისა და სქემო - ტექნიკური რეალიზაციის მიზნით, განვიხილოთ მათემატიკური აპარატი, რომელსაც საფუძვლად უდევს ალგორითმული (მიკროპროგრამული) ალგებრის სისტემის - ოპერატორული G(G1,G2,…) ალგებ-რის და პირობის P(α,β,…) ალგებრის ცნება, რომელთა ტერმინებშიც შეიძლება იყოს აღწერილი სხვა-დასხვა სახის ალგორითმული პროცესები.

აღნიშნული ალგებრები წარმოდგენილნი არიან ერთრეგისტრიანი ან მრავალრეგისტრიანი პერიოდულად განსაზღვრული გარდასახვების სახით, რომელთა ტერმინებშიც აღიწერება ზოგიერთი ალგორითმული პროცესების მიკროპროგრამები. ცნობილია, რომ ოპერატორული ალგებრის ელემენტებს უწოდებენ ოპერატორებს და განსაზღვრულნი არიან ისინი M ინფორმაციულ სიმრავლეზე, სადაც M იმ რეგისტრების მდგომარეობათა საერთო რიცხვია, რომლებიც მონაწილეობას ღებულობენ სისტემაში მიმდინარე გამოთვლით პროცესებში. დაუშვათ, რომ X^R ={…, x_{-1}^R, x_{-0}^R, x_1^R , …} (სადაც R=1,2,3,…) ორმხრივ უსასრულო რეგისტრების ერთობლიობაა და მათი ყოველი n-ური (-∞<n>∞) ელემენტი (ე.წ. ტრიგერი) ღებულობს ერთ-ერთ მნიშვნელობას სიმრავლიდან E_2 ={0,1}. ოპერატიული ალგებრის (როგორც ბაზურს, ასევე მათგან წარმოებულს) უწოდებენ ოპერატორებს, რომლებიც განისაზღვრებიან (ერთრეგისტრიანი ან მრავალრეგისტრიანი პერიოდულად განსაზრვრული) გარდასახვებით M სიმრავლისა M სიმრავლეში. პირობის ალგებრის ელემენტები (როგორც ბაზურს, ასევე მათგან წარმოებულს) განისაზღვრებიან M ინფორმაციულ სიმრავლეზე, როგორც პირობები, რომლებსაც შეუძლიათ მიიღონ ერთ-ერთი მნიშვნელობა თავისი სამი მნიშვნელობიდან **<T,F,U>,** (სადაც T-true, F-false, U-unknown). ოპერატორულ ალგებრაში ძირითად ოპერაციად მიღებულია გამრავლების ოპერაცია ანუ ოპერატორების თანმიმდევრული შესრულება, ხოლო პირობის ალგებრაში - ოპერაციები: კონიუნქცია, დიზიუნქცია, ინვერსია. განვიხილოთ ოპერაციები, რომელთა მეშვეობითაც ხორციელდება G,P ალგებრების ურთიერთდაკავშირება [1,3]:

1. α - დიზიუნქცია არის ოპერაცია, რომლის მიხედვითაც განისაზღვრება შესასრულებელი ოპერატორი ორი მოცემული ოპერატორიდან:

$Q = (_{\alpha}\ G_1 \cup G_2)$

სადაც $Q=G_1$ თუ α= true, ხოლო თუ α= false შესრულდება $Q=G_2$ ოპერატორი. α=unknowon -ს ეს არის შემთხვევა, რაც იწვევს error-ს. აღნიშნული ოპერაცია პროგრამირებაში ცნობილია, როგორც „პირობითი გადასვლის“ ოპერატორი ანუ ოპერატორი, რომელიც გამოიყენება განშტოებადი ალგორითმების სარეალიზაციოდ.

2. α - იტერაცია, არის ოპერაცია, რომლის მიხედვითაც განისაზღვრება შესასრულებელი ოპერატორის მრავალჯერადი გამეორება.

$Q = \{_{\alpha}\ G\}$

სადაც Q ოპერატორი ღებულობს G შესასრულებელი ოპერატორის მნიშვნელობებს მანამ, სანამ α= true. ოპერატორი Q არ არის განსაზღვრული თუ ლოგიკური პირობა α=unknowon -ს. იმ შემთხვევაში, თუ α= false ოპერატორი G არ სრულდება. α - იტერაციის ოპერაცია პროგრამირებაში გამოიყენება „ციკლური პროცესების“ სარეალიზაციოდ. აღწერილი ოპერაციების სახესხვაობები შეიძლება წარმოვადგინოთ შემდეგი გამოსახულებების სახით:

$Q=\{G\ \alpha\}=G\{_{\alpha}\ G\}$

$(G_1 \cup G_2)= (_{\alpha}\ G_2 \cup G_1)$

$Q = \{_F\ G\}=e$, სადაც e ცარიელი ოპერატორია.

β= G×α - არის ლოგიკური პირობა, რომელიც ღებულობს იმავე მნიშვნელობას რასაც α, ოღონდ G ოპერატორის შესრულების შემდეგ [3].

ცნობილია, რომ ნებისმიერი ოპერატორის წარმოდგენას ალგორითმული ალგებრის სისტემაში უწოდებენ ამ ოპერატორის რეგულიარულ მიკროპროგრამას[1,3]. მაგალითის სახით ქვემოთ

მოყვანილია რეგულიარული მიკროპროგრამა - Σ^c , რომლის შესრულების შედეგი ორი მთელი რიცხვის ჯამია:

$$\Sigma^c = \begin{matrix} 0^i & Z^i_{r1} \\ 0^j & Z^j_{r2} \end{matrix} \quad \Sigma_R{}^{ij} \qquad \textbf{(1)}$$

$$\Sigma_R{}^{i,j} = \left\{ \begin{matrix} \mathrm{mod}_2(X^i_n, X^j_n) \\ \alpha \quad \&(X^i_n, X^j_n) \end{matrix} \quad L^j_1 \right\} \qquad \textbf{(2)}$$

სადაც, $\Sigma_R{}^{i,j}$ -ორი მთელი რიცხვის (r1 და r2) შეკრების მიკროპროგრამაა. იგულისხმება, რომ რიცხვები r1 და r2 შესაბამისად შეტანილია X^i და X^j რეგისტრებში, ხოლო მიკროპროგრამის $\Sigma_R{}^{i,j}$ შესრულების შედეგი ფიქსირდება X^i რეგისტრში, იმ შემთხვევაში თუ R=i , ხოლო როცა R=j - X^j რეგისტრში.

0^R – X^R (R=i,j,..) რეგისტრის ნულოვან მდგომარეობაში გადაყვანის ოპერატორია, ხოლო $Z^R{}_r$ - r რიცხვის მნიშვნელობის X^R -რეგისტრში შეტანის ოპერატორია.

$\mathrm{mod}_2(X^i_n, X^j_n)$ – წარმოებული ლოგიკური ოპერატორია, მარეალიზებელი $f(x^i_n, x^j_n)$ გადამრთველი ფუნქციის $f(x^i_n, x^j_n)=\tilde{}x^i_n \& x^j_n \cup x^i_n \& \tilde{}x^j_n$ (ფუნქცია აღწერს X^i და X^j რეგისტრების n- ური თანრიგების (-∞<n>∞) მნიშვნელობების ორის მოდულით შეკრებას).

$\&(X^i_n, X^j_n)$ - ბაზური ლოგიკური ოპერატორია მარეალიზებელი $f(x^i_n, x^j_n)$ გადამრთველი ფუნქციის $f(x^i_n, x^j_n)=x^i_n \& x^j_n$ (ფუნქცია აღწერს X^i და X^j რეგისტრების n-ური თანრიგების მნიშვნელობების ლოგიკურ გამრავლებას - კონიუნქცია).

L^j_1 – X^j რეგისტრში შეტანილი რიცხვის ერთი თანრიგით მარცხნივ დაძვრის ოპერატორია.

α - ლოგიკური პირობაა, სადაც α= false, თუ ლოგიკური ოპერატორის $\&(X^i_n, X^j_n)$ შესრულების შედეგად X^j რეგისტრის ყველა ელემენტი მიიღებს ნულის მდგომარეობას, წინააღმდეგ შემთხვევაში α= true.

შევნიშნოთ, რომ მიკროპროგრამებში ერთ სვეტში შეტანილი ოპერატორები სრულდება პარალელურად. აღნიშნულიდან გამომდინარე იგულისხმება, რომ Z^i და Z^j, ასევე 0^i და 0^j ოპერატორები სრულდება ერთდროულად. ერთდროულად სრულდება აგრეთვე - $\mathrm{mod}_2(X^i_n, X^j_n)$ და $\&(X^i_n, X^j_n)$ ოპერატორები.

განვიხილოთ Σ^c მიკროპროგრამის შესრულების პროცედურა კონკრეტულ მაგალითზე. ვთქვათ, შესაკრებია ორი მთელი დადებითი რიცხვი: 87 და 78, რომელთა ჯამი უდრის 165. შევნიშნოთ, რომ 87 და 78, W და N სიმბოლოების კოდების მნიშვნელობებია შესაბამისად, ათობით ათვლის სისტემაში.

01010111 01001110	შეესაბამება რიცხვს 87, ორობით ათვლის სისტემაში შეესაბამება რიცხვს 78, ორობით ათვლის სისტემაში	X^1 X^2
00011001 01000110	ჯამი mod_2 &-კონიუნქცია (ლოგიკური ნამრავლი)	X^1 X^2
10001100	$L_1{}^2$ (X^2 –ის ერთი თანრიგით მარცხნივ დაძვრა)	X^2
00011001 10001100	ოპერანდების ფორმირება	X^1 X^2
10010101 10001000	ჯამი mod_2 &-კონიუნქცია (ლოგიკური ნამრავლი)	X^1 X^2
00010000	$L_1{}^2$ (X^2 –ის ერთი თანრიგით მარცხნივ დაძვრა)	X^2
10010101 00010000	ოპერანდების ფორმირება	X^1 X^2
10000101 00010000	ჯამი mod_2 &-კონიუნქცია (ლოგიკური ნამრავლი)	X^1 X^2
00100000	$L_1{}^2$ (X^2 –ის ერთი თანრიგით მარცხნივ დაძვრა)	X^1
10000101 00100000	ოპერანდების ფორმირება	X^1 X^2
10100101 00000000	ჯამი mod_2. (1*128+32+4+1=165) & - კონიუნქცია (ლოგიკური ნამრავლი)	X^1 X^2

რადგან &-კონიუნქციის შედეგი გახდა ნულის ტოლი. ცხადია, რომ მიკროპროგრამის შესრულება დამთავრებულია.

განვიხილოთ შეკრების ოპერაციის შებრუნებული ოპერაცია „გამოკლება“, კონკრეტულ მაგალითზე და შემდეგ შევადგინოთ „გამოკლების“ ოპერაციის მარეალიზებელი მიკროპროგრამა.

ვთქვათ გამოსათვლელია სხვაობა ორი (165 და 78) დადებით რიცხვებს შორის, რეზულტატი იქნება 165-78=87. აღნიშნული ოპერაციის შესასრულებლად საჭიროა მაკლები (78) გადავიყვანოთ შებრუნებულ კოდში და მიღებულ შედეგს ბოლო თანრიგში დაუმატოთ 1. აღნიშნული მანიპულაციების შესრულების შედეგად მიიღება მაკლები გადაყვანილი დამატებით კოდში. შედეგად გვექნება: ~(01001110)+00000001=10110010. აღწერილი პროცედურების შესრულების შემდეგ, ვასრულებთ მიკროპროგრამას „შეკრება“- - Σ^c.

10100101 10110010	საკლები, ანუ 165-ის ორობითი კოდი მაკლები, ანუ 78-ის დამატებითი კოდი	X^1 X^2
00010111 10100000	ჯამი mod_2 &-კონიუნქცია (ლოგიკური ნამრავლი)	X^1 X^2
01000000	L_1^2 (X^2 –ის ერთი თანრიგით მარცხნივ დაძვრა)	X^2
00010111 01000000	ჯამი mod_2 ოპერანდების ფორმირება &-კონიუნქცია (ლოგიკური ნამრავლი)	X^1 X^2
01010111 00000000	ჯამი mod_2 . (1*64+1*16+1*4+1*2+1=87) &-კონიუნქცია (ლოგიკური ნამრავლი)	X^1 X^2

რადგან &-კონიუნქციის (ლოგიკური ნამრავლის) შედეგი გახდა ნულის ტოლი ცხადია, რომ ოპერაციის შესრულება დამთავრებულია.

აღწერილი პროცედურის (ორი მთელი რიცხვის ოპერაცია „გამოკლება“: r1-r2) მარეალიზებელ მიკროპროგრამას - Σ^s აქვს შემდეგი სახე:

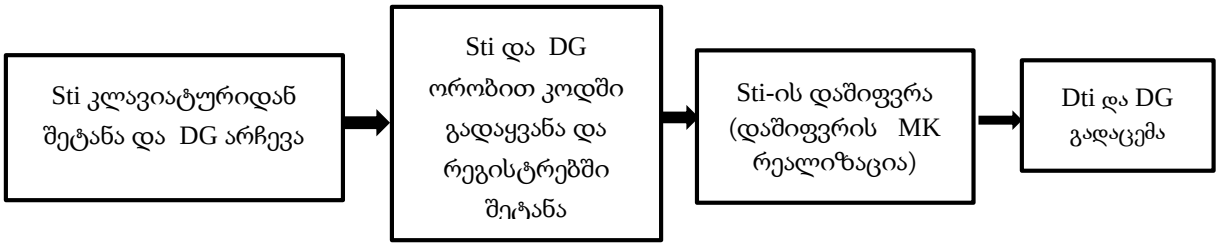
$$\Sigma^s = \begin{matrix} 0^i & Y^i_1 \\ 0^j & Z^j_{r2} \end{matrix} \quad {}^{\sim}X^j \Sigma_i^{i,j} \; 0^j \; Z^j_{r1} \; \Sigma_i^{i,j} \qquad (3)$$

სადაც, Y_1^j - არის ოპერატორი, რომლის შესრულების შედეგად X^i რეგისტრის ბოლო თანრიგი გადადის ერთის (ანუ - 0000....0001) მდგომარეობაში [3].

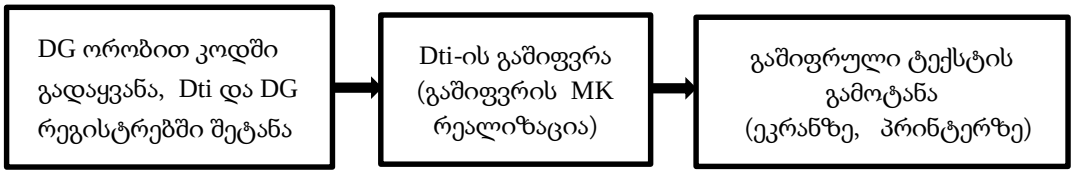
შემოთავაზებული მათემატიკური აპარატის გამოყენებით შედგენილია მიკროპროგრამები და მათი სქემური რეალიზაციის აბსტრაქტული მოდელები (იხ. ნახ. 1), კრიპტოგრაფიაში ფართოდ ცნობილი სიმეტრიული სისტემის მეთოდებისა, რომელთა რიცხვსაც მიეკუთვნება ცეზარის, ვიჟინერისა და ვერნამის მეთოდები.

მომავალში იგულისხმება, რომ Sti აიკრიფება კლავიატურიდან იმ სიმბოლოების გამოყენებით, რომლებიც განსაზღვრულია ASCII სტანდარტით. ამ სტანდარტის მიხედვით ნებისმიერი S_i სიმბოლოს რიცხვითი კოდის მნიშვნელობა, ათობით ათვლის სისტემაში, ეკუთვნის სასრულო სიმრავლეს R, სადაც R∈{32,33,……..,106,107}. ცხადია, რომ S_i სიმბოლოს რიცხვითი კოდი, რომ წარმოვადგინოთ ორობით ათვლის სისტემაში, საჭიროა შვიდი ბიტი. დაშიფვრა-გაშიფვრის პროცედურების განხორციელებისა და მათი მარეალიზებელი სქემის აღწერის მიზნით (იხ. ნახ. 1 და 2), განვიხილოთ სამი ორმხრივ უსასრულო ორობითი რეგისტრი X^1, X^2 და X^3 დავყოთ ეს რეგისტრები ტოლ ნაწილებად. თითოეულ ნაწილში N^q (q= 1, 2,3, …, KS) გავაერთიანოთ რეგისტრის რვა ელემენტი { x_7^q, x_6^q x_0^q} ანუ რვა ტრიგერი - T^q_j (სადაც, j=7,6,5,4,3,2,1,0). როგორც იყო აღნიშნული, თითოეულ სიმბოლოს თობით თვლის სისტემაში შეესაბამება კოდური რიცხვითი მნიშვნელობა KS_i , ხოლო ორობით კოდში KS_i კი რვა ბიტი b_j^q (j=7,6,…1,0). ცხადია, რომ ჯამური რაოდენობა ასეთი b_j^q ბიტებისა Σ= 8* KS სადაც , b_v (v=Σ,Σ-1, .. ,2,1,0). ნახ.1 ნაჩვენებია

b_v (T^q_j) აღწერილ მიკროპროგრამებში შესასრულებელი გარდასახვების (მიკროოპერაციების) მარეალიზებელი ლოგიკური სქემა, ხოლო ნახ. 2 წარმოდგენილია თითოეული ტრიგერის T^q_j (სადაც, j=7,6,5,4,3,2,1,0; q= 1, 2,3, …, KS) ბლოკური სახე.



ნახ.1. საწყისი ტექსტური ინფორმაციის დაშიფვრის პროცედურა.



ნახ.2. დაშიფრული ტექსტური ინფორმაციის გაშიფვრის პროცედურა.

X^1 ოეგისტოის N^q ხახილძი ძევიტახოთ დასაძიფოი ტექსტის $Ъ_z$ სიძბოლოს კოდის ორობითი მნიშვნელობა, ხოლო X^2 მდგომარეობა განისაზღვრება შემდეგი წესით (შევნიშნოთ, რომ X^3 რეგისტრი გამოიყენება შუალედური რეზულტატების დასამახსოვრებლად) :

ცეზარის მეთოდი. X^2 რეგისტრის ყოველ N^z (z=1,2,3,…) ნაწილში შეიტანება დამშიფრავი სიმბოლოს კოდის ორობითი მნიშვნელობა. აღნიშნულ მეთოდში დასაშიფრად, როგორც ცნობილია, გამოიყენება ერთი სიმბოლო.

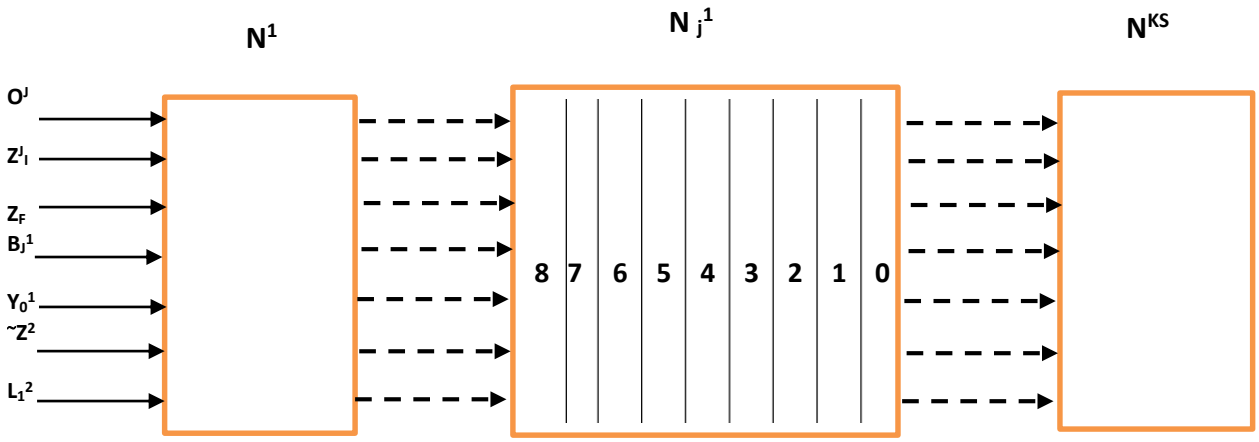
ვიჯინერის მეთოდი. როგორც ცნობილია, დამშიფრავი სიმბოლოების რაოდენობა აღნიშნულ მეთოდში სამის ტოლია. ვთქვათ ABC. ამ სიმბოლოებისგან ფორმირდება დამშიფრავი სტრიქონი (ABCABCABC .....), რომელშიც სიმბოლოების რაოდენობა ტოლია ან მეტია Sti-ში შემავალი სიმბოლოების რაოდენობაზე. ამგვარად ფორმირებული ტექსტის ყოველი სიმბოლოს რიცხვითი კოდი შეიცვლება შესაბამისი ორობითი კოდით და შეიტანება X^2 რეგისტრში.

ვერნამის მეთოდი. სტიდ-ი ირჩევს დამშიფრავ ტექსტს, რომელშიც სიმბოლოების რაოდენობა Sti-ში შემავალი სიმბოლოების რაოდენობის ტოლია. X^1 რეგისტრში Sti-ის შეტანის ანალოგიურად X^2 რეგისტრში შეიტანება დამშიფრავი ტექსტის სიმბოლოების კოდების ორობითი მნიშვნელობები. რეგისტრში ნაკლები სიმბოლოების შემცველი ტექსტის შეტანის შემდეგ, ამ რეგისტრის ბოლო თანრიგები იქნება შევსებული ნულებით [2].

განვიხილოთ ტექსტური ინფორმაციის დაშიფვრა - გაშიფვრის პროცედურების რეალიზ-აციის ორი ვარიანტი.

ვარიანტი 1. შევასრულოთ მიკროპროგრამა M^1:

$M^1 = mod_2(X^1_n, X^2_n)$, სადაც $(-\infty<n>\infty)$ (4)



ნახ.2. სალტეების მმართველი სიგნალები (იმპულსები).

რეალიზაციის შედეგად X^1 რეგისტრში დაფიქსირდება შიფროტექსტი. $mod_2(X_i^1, X_i^2)$ - წარმოებული ლოგიკური ოპერატორია (იხ. ფორმულა 2). თუ განმეორებით შევასრულებთ M^1 მიკროპროგრამას X^1 რეგისტრში დაფიქსირდება Sti.

ვარიანტი 2. X^1 რეგისტრის ყოველი N^z ნაწილისათვის თუ განვახორციელებთ Σ^c მიკროპროგრამის მარეალიზებელ გარდასახვებს და მიუთითებთ, რომ R=1, X^1 რეგისტრში დაფიქსირდება შიფროტექსტი. X^1 რეგისტრის მნიშვნელობა ასევე შეიტანება X^3 რეგისტრში.

ნახ.1.-ზე და ნახ.2.-ზე სალტეების წინ ჩაწერილი აღნიშვნები მიუთითებენ იმ მმართველ სიგნალებზე (იმპულსებზე), რომლებიც გამომუშავდებიან მართვის დროის დისკრეტულ მომენტში და რომლის სინქრონულადაც ხორციელდება ესა თუ ის (გარკვეული გარდასახვები რეგისტრებზე (რეგისტრზე). ასე მაგალითად, სიგნალები (იმპულსები) მიწოდებული სალტეებზე (0^j, Z_I^j და ა.შ.) შესაბამისად იწვევენ:

1. 0^j (j=1,2,3) X^j რეგისტრის ნულის მდგომარეობაში გადაყვანას;
2. Z^j_I – X^j რეგისტრის X^I - იურის მდგომარეობაში გადაყვანას;
3. $Z_f^{1,2}$ – $f(x_i^1,x_i^2)$ - გადამრთველი ფუნქციების $f(x_i^1, x_i^2) = \bar{x}_i^1 x_i^2 \vee x_i^1 \bar{x}_i^2$ და $f(x_i^1,x_i^2)=x_i^1-x_i^2$ მნიშვნელობების x^1 და x^2 რეგისტრებში შეტანას, შესაბამისად;
4. $B_j^1(b_j^2)$ - სიმბოლოს, რომელიც ეკუთვნის რეგისრებით $x^1(x^2)$ -ის q ნაწილის j ბიტს b_j ($b_j \cong E[0,1]$) რეგისტრში x^2 შეტანა;
5. Y_0^1 – X^1 , რეგისტრის ნულოვანი ტრიგერის ერთის შეტანა;
6. ~ Z^2 – X^2 რეგისტრის მდგომარეობის ინვენტირებას;
7. L_1^2 – X^2 რეგისტრში შეტანილი მნიშვნელობის მარცხნივ დაძვრას ერთი თანრიგით.

საწყისი ტექსტის აღდგენა Σ^s მიკროპროგრამის შესრულების შედეგი იქნება და იგი დაფიქსირდება X^1 რეგისტრში. ამ შემთხვევაში, ეტაპობრივად სრულდება შემდეგი გარდასახვები: პირველ ეტაპზე ხორციელდება ყოველი დამშიფრავი სიმბოლოს შესაბამისი კოდის დამატებით კოდში გადაყვანა და მისი დაფიქსირება X^2 რეგისტრში, შემდგომ X^3 რეგისტრის მნიშვნელობა გადაიწერება X^1 რეგისტრში და გაგრძელდება Σ^s მიკროპროგრამის შემადგენელი $\Sigma_1^{1,2}$ ოპერატორის შესრულება. <u>აღწერილი მიკროპროგამების (ცეზარის, ვიჯინერისა და ვერნამის მეთოდების) სქემო-ტექნიკური რეალიზაცია ნაჩვენებია ნახ. 3-ზე.</u>

ავღნიშნოთ, რომ კრიპტოგრაფიის სიმეტრიული სისტემების მეთოდების შემოთავაზებული მიკროპროგრამული რეალიზაცია, მიკროელექტრონიკაში თანამედროვე მიღწევების გათვალისწინებით არ უნდა წარმოადგენდეს დიდ სირთულეს და არ უნდა მოითხოვდეს დიდ დანახარჯებს. ანუ შეიძლება იყოს ეკონომიკური, ფინანსური თვალსაზრისითაც.

6.7. შებრუნებული მატრიცის მეთოდი

შებრუნებული მატრიცის მეთოდით, ტექსტური ინფორმაციის, დასაშიფრად და გასაშიფრად გამოიყენება მოცემული αA ალფავიტიდან, არა რომელიმე სიმბოლო ან სიმბოლოთა კრებული, როგორც ეს შესრულებული იყო §§1-3_აღწერილ მეთოდებში, არამედ ორი ორგანზომილებიანი DM (დამშიფრავი) და GM (გამშიფრავი) კვადრატული მატრიცა, რომელთა ყოველი წევრი განსაზღვრულია ნატურალურ მთელ რიცხვთა სიმრავლეზე. შევნიშნოთ, რომ შებრუნებული მატრიცის მეთოდით დაშიფვრისა და გაშიფვრის პროცედურების რეალიზაცია დაკავშირებულია რიგ რუტინულ მათემატიკურ გამოთვლებთან.

აქედან გამომდინარე, მოცემულ პარაგრაფში შემოთავაზებულია დაშიფვრისა და გაშიფვრის პროცედურების მარეალიზებელი ალგორითმების შედარებით გამარტივებული ვარიანტები, რაც იძლევა საშუალებას გამოთვლები შესრულდეს, როგორც პროგრამული, ასევე ემპირიული გზით. აგრეთვე, ერთმანეთთან შედარდეს და შემოწმდეს მიღებული შედეგები ალგორითმების რეალიზაციის სხვადასხვა ეტაპებზე. ამ მიზნით, შებრუნებული მატრიცის მეთოდით ტექსტური ინფორმაციის დაშიფვრისა და გაშიფვრის სადემონსტრაციო მაგალითებში განიხილება მხოლოდ სამი სტრიქონისა და სამი სვეტისაგან შემდგარი DM და GM მატრიცები (DM - დამშიფრავი ანუ საწყისი მატრიცა, რომელიც გამოიყენება TI დასაშიფრავად და GM - გამშიფრავი ანუ DM-ის შებრუნებული მატრიცა, რომელიც გამოიყენება შიფროტექსტის გასაშიფრად), სადაც:

$$DM = \begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{pmatrix}$$

$$GM = \begin{pmatrix} b_{00} & b_{01} & b_{02} \\ b_{10} & b_{11} & b_{12} \\ b_{20} & b_{21} & b_{22} \end{pmatrix}$$

ერთადერთი მოთხოვნა რაც უნდა იქნეს გათვალისწინებული DM მატრიცის არჩევისას არის ის, რომ მისი დეტერმინანტი (Det), რომელიც გამოითვლება ფორმულით (), არ უნდა იყოს ნულის ტოლი.

Det= a[0, 0]* a[1, 1]* a[2, 2] + a[2, 0]* a[0, 1]* a[1, 2]+ a[1, 0]* a[2, 1]* a[0, 2] –
-(a[0, 2]* a[1, 1]* a[2,0]+ a[0, 0]* a[2, 1]* a[1, 2]+ a[1, 0]* a[0, 1]* a[2, 2])

თუ DM მატრიცის დეტერმინანტი Det#0, მაშინ არსებობს მისი შებრუნებული მატრიცა, მოცემულ შემთხვევაში GM მატრიცა, რომლის წევრები გამოითვლება შემდეგნაირად:

b[0, 0] = a[1, 1] * a[2, 2] - a[1, 2] * a[2, 1];
b[0, 1] = a[0, 2] * a[2, 1] - a[0, 1] * a[2, 2];
b[0, 2] = a[0, 1] * a[1, 2] - a[0, 2] * a[1, 1];

b[1, 0] = a[1, 2] * a[2, 0] - a[1, 0] * a[2, 2];
b[1, 1] = a[0, 0] * a[2, 2] - a[0, 2] * a[2, 0];
b[1, 2] = a[0, 2] * a[1, 0] - a[0, 0] * a[1, 2];

b[2, 0] = a[1, 0] * a[2, 1] - a[1, 1] * a[2, 0];
b[2, 1] = a[0, 1] * a[2, 0] - a[0, 0] * a[2, 1];
b[2, 2] = a[0, 0] * a[1, 1] - a[0, 1] * a[1, 0];

ცნობილია , რომ თუ Det#0 , მაშინ

$$DM \times GM = \begin{pmatrix} e_{00} & 0 & 0 \\ 0 & e_{11} & 0 \\ 0 & 0 & e_{22} \end{pmatrix}$$

მიღებულ მატრიცაში, მთავარ დიაგონალზე განლაგებული ელემენტების მნიშვნელობები ერთმანეთის ტოლია, ანუ $e_{00}=e_{11}=e_{22}$, ავღნიშნოთ K სიმბოლოთი <u>და განსაზღვრულნი არიან ინტერვალზე (-∞<K>∞).</u> ფორმულა (), შეიძლება ჩაიწეროს შემდეგი სახით:

$$\begin{pmatrix} e_{00} & 0 & 0 \\ 0 & e_{11} & 0 \\ 0 & 0 & e_{22} \end{pmatrix} = K \times \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = K \times E$$

E მატრიცის ელემენტები გამოითვლება:

e[0, 0] = a[0,0]*b[0,0]+a[0,1]*b[1,0]+a[0,2]*b[2,0]
e[0, 1] = a[0,0,]*b[0,1]+a[0,1]*b[1,1]+a[0,2]*b[2,1]
e[0, 2] = a[0,0]*b[0,2]+a[0,1]*b[1,2]+a[0,2]*b[2,2]

e[1, 0] = a[1,0]*b[0,0]+a[1,1]*b[1,0]+a[1,2]*b[2,0]
e[1, 1] = a[1,0]*b[0,1]+a[1,1]*b[1,1]+a[1,2]*b[2,1]
e[1, 2] = a[1,0]*b[0,2]+a[1,1]*b[1,2]+a[1,2]*b[2,2]

e[2, 0] =a[2,0]*b[0,0]+a[2,1]*b[1,0]+a[2,2]*b[2,0]
e[2, 1] = a[2,0]*b[0,1]+a[2,1]*b[1,1]+a[2,2]*b[2,1]
e[2, 2] = a[2,0]*b[0,2]+a[2,1]*b[1,2]+a[2,2]*b[2,2]

განვიხილოთ დაშიფვრისა და გაშიფვრის შესრულების პროცედურების რეალიზაციის ძირითადი ეტაპები.

TI-ის დასაშიფრად საჭიროა:

1. შემოწმდეს დასაშიფრი TI-იის (TI={S_i} სადაც i=0,1,2,…,Z), ანუ სტრიქონის სიგრძე, რომელიც უნდა აკმაყოფილებდეს პირობას :

(Z+1)%3=0,

წინააღმდეგ შემთხვევაში, თუ () პირობა არ სრულდება მაშინ, TI-იის სტრიქონს ბოლოში უნდა დაემატოს ერთი ან ორი სიმბოლო, ვთქვათ სიმბოლო @ , რათა დაკმაყოფილდეს პირობა

(Z_1+1)%3=0

სადაც Z_1 ახლად ფორმირებული TI-იის სტრიქონის სიგრძეა (Z_1 >= Z).

2. αA-ალფავიტიდან (იხ. ცხრ. 1-3) ყოველ S_i სიმბოლოს მიენიჭოს მისი შესაბამისი კომპიუტერული კოდი. შედეგად მიიღება TI-იის კოდური სტრიქონი:

TIKS ={ S_i^k} სადაც i=0,1,2,…,Z_1

3. TIKS შემავალი კოდების მნიშვნელობები დაჯგუფდეს დაწყებული მარცხნიდან (S_0^k კოდის მნიშვნელობიდან) მარჯვნივ ტეტრადებად (ბლოკებად). მიღებულ მიმდევრობაში:

T_0, T_1, T_2,…., T_t

T_j (j=0,1,2,..,t) -არის ბლოკი (ტეტრადა), t - ამ ბლოკების საერთო რაოდენობა.

თითოეულ ბლოკში გაერთიანებულია ყოველი მომდევნო სამი სიმბოლოს შესაბამისი კოდური მნიშვნელობები. კერძოდ, { S_{3*j}^k , S_{3*j+1}^k, S_{3*j+2}^k}.

4. ყოველ T_j ={ S_{3*j}^k , S_{3*j+1}^k, S_{3*j+2}^k} ბლოკში შემავალი დასაშიფრი TI-იის სიმბოლოს კოდური მნიშვნელობისათვის მოიძებნება ბლოკი U_j={ D_{3*j}^k , D_{3*j+1}^k, D_{3*j+2}^k} შიფროტექსტის სიმბოლოს კომპიუტერული კოდის მნიშვნელობა შემდეგი გამოთვლების გზით:

$D_{3*j}^k = S_{3*j}^k * a[0, 0] + S_{3*j+1}^k * a[1, 0] + S_{3*j+2}^k * a[2, 0]$

$D_{3*j+1}^k = S_{3*j}^k * a[0, 1] + S_{3*j+1}^k * a[1, 1] + S_{3*j+2}^k * a[2, 1]$

$D_{3*j+2}^k = S_{3*j}^k * a[0, 2] + S_{3*j+1}^k * a[1, 2] + S_{3*j+2}^k * a[2, 2]$

სადაც, a[*, *] – DM მატრიცის ელემენტებია.

ასეთი წესით გამოთვლილი U_j , j =0,1,..., t ბლოკების მიმდევრობა U_0 , U_1 ,..., U_t არის დაშიფრულ TI-ში (შიფროტექსტი) შემავალი სიმბოლოების კოდური მნიშვნელობები, ShifTIK= {D_i^k}, რომლებსაც βA ალფავიტში ცალსახად შეესაბამება გარკვეული სიმბოლო - D_i , მათი მიმდევრობა კი ქმნის შიფროტექსტს - ShifTI= {D_i}.

ShifTI გასაშიფრავად საჭიროა:

5. ყოველი U_j , j =0,1,..., t ბლოკში შემავალი კოდური მნიშვნელობების მიხედვით და GM მატრიცის გამოყენებით, შესაბამისად გამოვთვალოთ T_j ={ S_{3*j}^k , S_{3*j+1}^k, S_{3*j+2}^k} ბლოკში შემავალი დასაშიფრი TI-იის სიმბოლოს კოდური მნიშვნელობი, შემდეგი წესით:

$S_{3*j}^k = (D_{3*j}^k * b[0, 0] + D_{3*j+1}^k * b[1, 0] + D_{3*j+2}^k * b[2, 0]):K$

$S_{3*j+1}^k = (D_{3*j}^k * b[0, 1] + D_{3*j+1}^k * b[1, 1] + D_{3*j+2}^k * b[2, 1]):K$

$S_{3*j+2}^k = (D_{3*j}^k * b[0, 2] + D_{3*j+1}^k * b[1, 2] + D_{3*j+2}^k * b[2, 2]):K$

სადაც, b[*, *] – GM მატრიცის ელემენტებია.

()_ის შედეგად ჩატარებული გამოთვლებით მიიღება საწყის TI-ში შემავალი სიმბოლოების კოდური მნიშვნელობები

TIKS ={ S_i^k} სადაც i=0,1,2,…,Z_1

რომლებსაც თავის მხრივ αA -ალფავიტიდან (იხ. ცხრ. 1-3) ცალსახად შეესაბამებიან შესაბამისი სიმბოლოები ანუ მიიღება ახლად ფორმირებული TI, რომლიდანაც პირველი Z სიმბოლო ეკუთვნის საწყის დასაშიფრ TI და რომელზეც განხორციელდა ყველა ზემოთ აღწერილი მანიპულაციები.

განვიხილოთ კონკრეტულ მაგალითზე ტექსტური ინფორმაციის შებრუნებული მატრიცის მეთოდით დაშიფვრისა და გაშიფვრის აღწერილი პროცედურები.

ავიღოთ TI-ის დასაშიფრი DM მატრიცა:

$$DM = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 3 & 1 \\ 2 & 1 & 0 \end{pmatrix}$$

დავრწმუნდეთ, რომ არსებობს DM მატრიცის შებრუნებული მატრიცა. გამოვთვალოთ დეტერმინანტი ფორმულა () გამოყენებით:

Det=1*3*0+2*2*1+0*1*3-(2*3*3+1*1*1+0*2*0)=0+4+0-(18+1+0)=4-19=-15≠0 .

მაშასადამე, აღებული DM მატრიცის დეტერმინანტი რადგან ≠ 0-ს, ე.ი. აქვს შებრუნებული GM გამშიფრავი მატრიცა. აღნიშნული კი ფორმულა ()-ს საფუძველზე გამოითვლება და მიიღება შემდეგი სახის:

$$\mathrm{GM} = \begin{pmatrix} 3*0-1*1 & 3*1-2*0 & 2*1-3*3 \\ 1*2-0*0 & 1*0-2*3 & 3*0-1*1 \\ 0*1-3*2 & 2*2-1*1 & 1*3-2*0 \end{pmatrix} = \begin{pmatrix} -1 & 3 & -7 \\ 2 & -6 & -1 \\ -6 & 3 & 3 \end{pmatrix}$$

ვიპოვოთ k კოეფიციენტი :

$$\mathrm{DM} * \mathrm{GM} = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 3 & 1 \\ 2 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} -1 & 3 & -7 \\ 2 & -6 & -1 \\ -6 & 3 & 3 \end{pmatrix} =$$

$$\begin{pmatrix} 1*(-1)+2*2+3*(-6) & 1*3+2*(-6)+3*3 & 1*(-7)+2*(-1)+3*3 \\ 0*(-1)+2*3+1*(-6) & 0*3+3*(-6)+1*3 & 0*(-7)+3*(-1)+1*3 \\ 2*(-1)+1*2+0*(-6) & 2*3+1*(-6)+0*3 & 2*(-7)+1*(-1)+0*3 \end{pmatrix} =$$

$$\begin{pmatrix} -15 & 0 & 0 \\ 0 & -15 & 0 \\ 0 & 0 & -15 \end{pmatrix} = (-15) \times \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = k \times E, \quad \mathrm{k} = -15$$

ავიღოთ დასაშიფრი TI-ია: „GEORGIA“.

1. TI={S_i}, სადაც i=(0,1,2,…,Z), სტრიქონის სიგრძე არ აკმაყოფილებს მოთხოვნას (Z+1)%3=0, ამიტომ უნდა დაემატოს რაიმე სიმბოლო ვთქვათ „?“. იმისათვის, რომ დაკმაყოფილდეს () მოთხოვნა, საჭიროა ორი სიმბოლოს დამატება, ვთქვათ - „??“. ფორმირებული Z მიიღებს Z_1 ფორმას Z_1= GEORGIA??

2. Z_1-ის თითოეულ სიმბოლოს შეუსაბამოთ კონკრეტული კომპიუტერული კოდები ცხრილი (1-3)-ის გამოყენებით - 72,70,80,83,72,74,66,64,64.

3. TIKS შემავალი კოდების მნიშვნელობები დავყოთ სამ-სამი სიმბოლოსაგან შემდგარი ბლოკებად, ბლოკების რაოდენობა t=3:

T_0 (S_{3*0}^k) = 72,70,80, T_1 (S_{3*1}^k) = 83,72,74, T_2 (S_{3*2}^k) = 66,64,64.

4. ბლოკში შემავალი კოდების დასაშიფრად ვიქცევით შემდეგნაირად: DM მატრიცის ელემენტები მრავლდება T_0, T_1, T_2 ბლოკის ელემენტებზე, მიიღება შიფროტექსტი:

$$D_{3*0}^k = T_0\,(S_{3*0}^k) \times \mathrm{DM} = (72\ 70\ 80) \times \begin{pmatrix} 1 & 2 & 3 \\ 0 & 3 & 1 \\ 2 & 1 & 0 \end{pmatrix} = (232\ 434\ 286) = U_0$$

$$D_{3*1}^k = T_1\,(S_{3*1}^k) \times \mathrm{DM} = (83\ 72\ 74) \times \begin{pmatrix} 1 & 2 & 3 \\ 0 & 3 & 1 \\ 2 & 1 & 0 \end{pmatrix} = (231\ 456\ 321) = U_1$$

$$D_{3*2}^k = T_2\,(S_{3*2}^k) \times \mathrm{DM} = (66\ 64\ 64) \times \begin{pmatrix} 1 & 2 & 3 \\ 0 & 3 & 1 \\ 2 & 1 & 0 \end{pmatrix} = (194\ 388\ 262) = U_2$$

5. მიღებული U_0, U_1, U_2 შიფროტექსტიდან GM მატრიცის გამოყენებით (ელემენტებზე გამრავლებით) და K კოეფიციენტზე გაყოფით, მიიღება TI-ის კომპიუტერული კოდები:

$$(U_0 \times \mathrm{GM})/K = ((232\ 434\ 286) \times \begin{pmatrix} -1 & 3 & -7 \\ 2 & -6 & -1 \\ -6 & 3 & 3 \end{pmatrix})/(-15) = (-1080\ -1050\ -1200)/\ (-15) = (72\ 70\ 80) = S_{3*0}^k$$

$$(U_1 \times \mathrm{GM})/K = ((231\ 456\ 321) \times \begin{pmatrix} -1 & 3 & -7 \\ 2 & -6 & -1 \\ -6 & 3 & 3 \end{pmatrix})/(-15) = (-1245\ -1080\ -1110)/\ (-15) = (83\ 72\ 74) = S_{3*1}^k$$

$$(U_2 \times GM)/K = ((194\ 388\ 262) \times \begin{pmatrix} -1 & 3 & -7 \\ 2 & -6 & -1 \\ -6 & 3 & 3 \end{pmatrix})/(-15) = (-990\ -960\ -960)/\ (-15) = (66\ 64\ 64) = S_{3*2}^{k}$$

მაშასადამე, მიღებული კომპიუტერული კოდების S_{3*0}^{k}, S_{3*1}^{k}, S_{3*2}^{k} ცხრილი (1-3)-ს შესაბამისობით კი, ფორმირდება დასაშიფრი TI-ია.

განვიხილოთ კიდევ ერთი ვარიანტი TI დაშიფვრისა და გაშიფვრის პროცედურების რეალიზაციის შებრუნებული მატრიცის მეთოდით საოფისე პროგრამის MS Excel-ის ფუნქციების გამოყენებით, რომელიც ნაჩვენებია ნახ. , სადაც

ა) დეტერმინანტი (Det=-15) გამოთვლილია საწყისი მატრიცის DM მონაცემების მიხედვით ფუნქციით: MDETERM().

ბ) შებრუნებული მატრიცა გამოთვლილია ფუნქციით : MINVERSE().

გ) ერთეულოვანი მატრიცა გამოთვლილია ფუნქციით : MMULT().

დ) მაგალითის სახით ნაჩვენებია TI="KekELiA" დაშიფვრისა და გაშიფვრის პროცედურული რეალიზაცია, კერძოდ:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
2															
3		საწყისი მატრიცა													
4		1	2	3											
5	DM=	0	3	1											
6		2	1	0											
7															
8		შებრუნებული მატრიცა													
9		0.066667	-0.2	0.466666667											
10	GM=	-0.13333	0.4	0.066666667											
11		0.4	-0.2	-0.2											
12															
13		ერთეულოვანი მატრიცა													
14		1	-1.11022E-16	-1.11022E-16											
15	E=	0	1	0											
16		0	0	1											
17															