

საქართველოს ტექნიკური უნივერსიტეტი

თ. ბახტაძე

ამოცანათა კრებული
C# უემსჯავლელთათვის



რეგისტრირებულია სტუ-ს
სარედაქციო-საგამომცემლო
საბჭოს მიერ

თბილისი
2008

ეს ამოცანათა კრებული ამჟამად ყველაზე თანამედროვე ოპერატორული ენის "სი შარპის" შემსწავლელთათვისაა განკუთვნილი. საკითხები დაყოფილია სამ ნაწილად, რაც ჩვეულებრივ გულისხმობს სამსემესტრიან კურსს. პირველი ნაწილი შეიძლება გამოყენებულ იქნას სხვა ოპერატორული ენებისთვისაც, ხოლო დარჩენილი ორი ნაწილი - მხოლოდ ობიექტზე ორიენტირებული ენებისთვის. უნდა აღინიშნოს მოყვანილი ამოცანების ალგორითმული სიმარტივე. სავარაუდოდ იმიტომ, რომ კრებული უპირატესად განკუთვნილია ენის შესწავლისათვის და არა ალგორითმული აზროვნების დახვეწისათვის. იგი გამოდგება, როგორც პროგრამირების შესწავლის დამწყებთათვის, ასევე მათთვის, ვისაც გამოცდილება აქვს სხვა, როგორც ობიექტზე ორიენტირებულ, ისე ობიექტზე არაორიენტირებულ ენებში. კრებული გარკვეულად მოიცავს ენის ყველა ასპექტს. მეორე მხრივ, ეს ენა შედის Visual Studio Net სისტემის შემადგენლობაში, რომლის შესაძლებლობებზე ამოცანები განხილულია მინიმალურად, ვინაიდან სცილდება კრებულის ფარგლებს.

კრებულში ასევე მოყვანილია მაგალითები, რომლებიც თვალსაჩინოს ხდის, თუ რა მოეთხოვება მოსწავლეს. პროგრამების გაშვება შესაძლებელია როგორც Visual Studio Net-ის, ასევე Net FrameWork სისტემების საშუალებით. კრებული დახმარებას გაუწევს სტუდენტს ენის ღრმა ათვისებაში.

კრებული განკუთვნილია სტუდენტებისა და მაგისტრანტებისთვის, რომელთაც სურთ შეისწავლონ და გააღრმავონ ობიექტზე ორიენტირებული ენის საფუძვლები.

რეცენზენტი

პროფესორი ა. ფრანგიშვილი

© საგამომცემლო სახლი „ტექნიკური უნივერსიტეტი“, 2008

ISBN 978-9941-14-097-6

<http://www.gtu.ge/publishinghouse/>



სარჩევი

შესავალი.....	4
პირველი ნაწილი	5
მეორე ნაწილი	21
მესამე ნაწილი.....	56
ლიტერატურა.....	80

შესავალი

ობიექტზე ორიენტირებული პროგრამირების კონცეფციის შესწავლა ძალზე მნიშვნელოვანია, ვინაიდან მასზეა დაფუძნებული თანამედროვე პროგრამული პროდუქტები. ეს კონცეფცია უფრო ნათელს ხდის სამომხმარებლო პროგრამებს და საშუალებას იძლევა იოლად გადავერთოთ ობიექტზე ორიენტირებული ნებისმიერი ენის გამოყენებაზე. C# (სი შარპი) წარმოადგენს ობიექტზე ორიენტირებული პროგრამირების ერთ-ერთ ყველაზე თანამედროვე ენას [1-7]. C# დამუშავებულია კორპორაცია Microsoft-ში და აქტიურად ვითარდება, როგორც Visual Studio Net-ის ერთ-ერთი შემადგენელი ნაწილი. ამ ენის საფუძვლიანი ათვისება საშუალებას იძლევა უფრო ღრმად გავვერკვეთ ნებისმიერ თანამედროვე პროგრამულ პროექტში. კრებული შედგება სამი ნაწილისაგან და მოიცავს 250 საკითხს. საკითხების მრავალფეროვნება ინდივიდუალური დავალების მიცემის საშუალებას იძლევა. იგი ასევე შეიძლება გამოყენებულ იქნას ტესტირებისთვის. კრებულში მოცემული ამოცანები განკუთვნილია ენის როგორც არაობიექტური, ასევე ობიექტური ნაწილებისათვის. ამის გამო იგი გამოდგება როგორც დამწყებთათვის, ასევე გარკვეული გამოცდილების მქონე სტუდენტებისთვის.

პირველი ნაწილი

1. მოცემულ სტრიქონში, რომელიც შეგვყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ, s-ცვლადს, ვიპოვოთ: თქვენს სახელში შემავალი ასოების რაოდენობა(თუ ასეთი არსებობს);
2. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: წევრების ჯამი;
3. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq -R \\ \sqrt{R^2 - x^2}, & 0 < x \leq R \\ 0, & x > R \end{cases}$$

4. მოცემულ სტრიქონში, რომელიც შეგვყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს, ვიპოვოთ: რამდენჯერ შედის თქვენი სახელი ამ სტრიქონში;
5. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: დადებითი წევრების ჯამი;
6. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} h, & x \leq -a \\ \frac{h}{a^2} \times x^2, & 0 < x \leq a \\ h, & x > a \end{cases}$$

7. მოცემულ სტრიქონში, რომელიც შეგვყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს, ვიპოვოთ: არსებობს თუ არა თქვენი სახელის მსგავსი თანმიმდევრობა ამ სტრიქონში (ანუ სახელის ასოებს შორის შესაძლებელია სხვა სიმბოლოებიც იყოს);

8. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: უარყოფითი წევრების ჯამი;
9. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{h}{a} \times x, & -a < x \leq 0 \\ a \\ -\frac{h}{a} \times x, & 0 > x \leq a \\ 0, & x > a \end{cases}$$

10. მოცემულ სტრიქონში, რომელიც შეგვყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს, გამოთვალეთ სტრიქონის სიგრძე ფუნქციის გამოყენების გარეშე ;
11. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: ლუწინდექსებიანი წევრების ჯამი;
12. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{c+h}{a^2} \times (x-a)^2 - h, & 0 < x < \pi \\ 0, & x \geq \pi \end{cases}$$

13. მოცემულ სტრიქონში, რომელიც შეგვყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს, ვიპოვოთ: სტრიქონის ბოლოდან რომელ პოზიციაშია თქვენი სახელი (თუკი ასეთი არსებობს);
14. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: კენტინდექსებიანი წევრების ჯამი;
15. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq -\frac{\pi}{2} \\ \cos x, & -\frac{\pi}{2} < x < \frac{\pi}{2} \\ 0, & x \geq \frac{\pi}{2} \end{cases}$$

16. მოცემულ სტრიქონში, რომელიც შეგვეყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს, ვიპოვოთ: რომელი პოზიციებიდან შედის თქვენი სახელი ამ სტრიქონში;
17. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: ლუწინდექსებიანი დადებითი წევრების ჯამი;
18. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{c+h}{a^2} \times (x-a)^2 - h, & 0 < x \leq a \\ 0, & x > a \end{cases}$$

19. მოცემულ სტრიქონში, რომელიც შეგვეყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. შეცვალეთ ყოველი თქვენი სახელი რომელიმე, მაგალითად, თქვენი მეგობრის სახელით;
20. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: კენტინდექსებიანი დადებითი წევრების ჯამი;
21. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} h, & x \leq -a \\ e^x + h, & -a < x < 0 \\ e^{-x} + h, & 0 \leq x < a \\ h, & x \geq a \end{cases}$$

22. მოცემულ სტრიქონში, რომელიც შეგვეყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. დააღაგეთ სიტყვები უკუმიმდევრობით;
23. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: ლუწინდექსებიანი უარყოფითი წევრების ჯამი;
24. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{2h}{a}x, & -0 < x < \frac{a}{2} \\ -\frac{2h}{a}x, & \frac{a}{2} \leq x < a \\ \frac{2h}{a}x, & a \leq x < \frac{3a}{2} \\ 0, & x \geq \frac{3a}{2} \end{cases}$$

25. მოცემულ სტრიქონში, რომელიც შეგვეყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. დააღაგეთ ასოები უკუმიმდევრობით;
26. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: კენტინდექსებიანი უარყოფითი წევრების ჯამი;
27. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq 0 \\ \operatorname{tg}x, & 0 < x \leq \frac{\pi}{3} \\ h, & \frac{\pi}{3} \leq x \leq a \\ -\frac{h}{b}x, & a < x \leq a+b \\ 0, & x > a+b \end{cases}$$

28. მოცემულ სტრიქონში, რომელიც შეგვეყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. დაალაგეთ სიტყვები ანბანის მიხედვით;
29. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: წევრების რაოდენობა პირველ ნულოვან წევრამდე;
30. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{h}{a^2} x^2, & 0 < x < a \\ \frac{h}{a^2} (x-2a)^2, & a \leq x < 3a \\ \frac{h}{a^2} (x-5a)^2, & 3a \leq x < 5a \\ 0, & x \geq 5a \end{cases}$$

31. მოცემულ სტრიქონში, რომელიც შეგვეყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. შეცვალეთ ყოველი ასო X-ით;
32. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: დადებითი წევრების რაოდენობა;
33. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{c+h}{a^2} \times (x-a)^2 - h, & 0 < x \leq a \\ 0, & x > a \end{cases}$$

34. მოცემულ სტრიქონში, რომელიც შეგვყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ. s-ცვლადს. სიტყვებს შორის დატოვებთ მხოლოდ ერთი შუალედის სიმბოლო;
35. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: უარყოფითი წევრების რაოდენობა;
36. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq -\frac{\pi}{2} \\ \cos x, & -\frac{\pi}{2} < x \leq \frac{\pi}{2} \\ \cos(x - \pi), & \frac{\pi}{2} < x \leq \frac{3}{2}\pi \\ \cos(x - 2\pi), & \frac{3}{2}\pi < x \leq \frac{5}{2}\pi \\ 0, & x > \frac{5}{2}\pi \end{cases}$$

37. მოცემულ სტრიქონში, რომელიც შეგვყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. სიტყვებს შორის შუალედი შეცვალეთ სიმბოლო Z-ით;
38. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: მაქსიმალური ელემენტი;
39. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq -R \\ \sqrt{R^2 - x^2}, & -R < x \leq R \\ 0, & x > R \end{cases}$$

40. მოცემულ სტრიქონში, რომელიც შეგვეყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. თითოეული სიტყვა მოაქციეთ X და Z სიმბოლოებს შორის;
41. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: მინიმალური ელემენტი მოდულით;
42. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x < a \\ 0, & x \neq n \times a \\ h, & x = n \times a, n = 1, 2, 3, \dots \end{cases}$$

43. მოცემულ სტრიქონში, რომელიც შეგვეყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ. s-ცვლადს. შეიყვანეთ ან დააფორმირეთ რაიმე ახალი სტრიქონი და მიუწერეთ მოცემულ სტრიქონს მარცხნიდან;
44. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: მაქსიმალური ელემენტი მოდულით;
45. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & -x \leq a \\ \frac{h}{a} \times (x + a) - h, & -a < x \leq 0 \\ \cos x, & 0 < x \leq \frac{\pi}{2} \\ \frac{h}{a} (x - \pi), & \frac{\pi}{2} < x \leq \pi \\ \cos(x - \pi), & \pi < x \leq \frac{3}{2} \pi \\ 0, & x > \frac{3}{2} \pi \end{cases}$$

46. მოცემულ სტრიქონში, რომელიც შეგვეყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. შეიყვანეთ ან დააფორმირეთ რაიმე ახალი სტრიქონი და ჩასვით მოცემული სტრიქონის შუაში;
47. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: კენტინდექსებიან ელემენტთა შორის უდიდესი;
48. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq 0 \\ a, & (2n-1)a < x \leq 2na \\ 0, & 2na < x \leq (2n+1)a, n = 1, 2, 3, \dots \end{cases}$$

49. მოცემულ სტრიქონში, რომელიც შეგვეყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. შეიყვანეთ ან დააფორმირეთ რაიმე ახალი სტრიქონი და მიუწერეთ მოცემული სტრიქონის პირველი სიტყვა მარჯვნიდან, ხოლო მეორე - მარცხნიდან;
50. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: ლუწინდექსებიან ელემენტთა შორის უდიდესი;
51. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq 0 \\ \sin x, & 0 < x \leq b \\ e^{-(x-\pi/2)}, & b < x \leq d \\ \frac{\sin(b)}{a}(x-d)x, & d < x \leq d+a \\ 0, & x > d+a \end{cases}$$

52. მოცემულ სტრიქონში, რომელიც შეგვყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. შეიყვანეთ ან დააფორმირეთ რაიმე ახალი სტრიქონი და მიუწერეთ მოცემული სტრიქონის ბოლო სიტყვა მარჯვნიდან;
53. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: ნულოვანი ელემენტების რაოდენობა;
54. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} b, & x \leq -a \\ \frac{h}{a^2} \times x^2 - h, & -a < x \leq a \\ b, & x > a \end{cases}$$

55. მოცემულ სტრიქონში, რომელიც შეგვყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. შეიყვანეთ ან დააფორმირეთ რაიმე ახალი სტრიქონი და მიუწერეთ მოცემულ სტრიქონს მარჯვნიდან და მარცხნიდან;
56. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: 1-ის ტოლ ელემენტთა რაოდენობა;
57. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{-h}{a^2} \times x^2 + h, & -a < x \leq a \\ 0, & x > a \end{cases}$$

58. მოცემულია სტრიქონი, რომელიც შეგვყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. შეიყვანეთ ან დააფორმირეთ რაიმე ახალი სტრიქონი და დაადგინეთ, ემთხვევა თუ არა მასში შემავალი სიტყვები მოცემული სტრიქონის სიტყვებს;
59. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: წევრების ნამრავლი;
60. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{-h}{a} \times (x+a), & -a < x \leq 0 \\ \frac{h}{a} \times (x-a), & 0 < x \leq a \\ \sin(x-a-\frac{\pi}{2}), & a < x \leq a+\pi \\ 0, & x > a+\pi \end{cases}$$

61. მოცემულ სტრიქონში, რომელიც შეგვყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. შეიყვანეთ ან დააფორმირეთ რაიმე ახალი სტრიქონი და მიუწერეთ მისი პირველი სიტყვა მოცემულ სტრიქონს მარჯვნიდან;
62. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: დადებითი წევრების ნამრავლი;
63. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{d-h}{a^2} \times x^2 - d, & -a < x \leq a \\ 0, & x > a \end{cases}$$

64. მოცემულ სტრიქონში, რომელიც შეგვყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. შეიყვანეთ ან დააფორმირეთ რაიმე ახალი სტრიქონი და მიუწერეთ მისი პირველი სიტყვა მოცემულ სტრიქონს მარცხნიდან;
65. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: უარყოფითი წევრების ნამრავლი;
66. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} he^{-a}, x \leq -a \\ he^x, -a < x \leq 0 \\ he^{-x}, 0 < x \leq a \\ e^{-a}, x > a \end{cases}$$

67. მოცემულ სტრიქონში, რომელიც შეგვეყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. შეიყვანეთ ან დააფორმირეთ რაიმე ახალი სტრიქონი და მიუწერეთ მისი ბოლო სიტყვა მოცემულ სტრიქონს მარჯვნიდან;
68. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: ლუწინდექსიანი ელემენტების ნამრავლი;
69. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} h, x \leq -a \\ -h, -a < x \leq a \\ h, a < x \leq 2a \\ \frac{d}{c}(x-2a), 2a < x \leq 2a+c \\ 0, x > 2a+c \end{cases}$$

70. მოცემულ სტრიქონში, რომელიც შეგვეყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. შეიყვანეთ ან დააფორმირეთ რაიმე ახალი სტრიქონი და მიუწერეთ მისი ბოლო სიტყვა მოცემულ სტრიქონს მარცხნიდან;
71. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: მოცემული რიცხვების კვადრატების ჯამი;
72. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{-h}{a} \times (x+a), & -a < x \leq 0 \\ \frac{h}{a} (x-a), & 0 < x \leq 2a \\ \operatorname{tg}(x-2a), & 2a < x \leq 3a \\ h, & x > 3a \end{cases}$$

73. მოცემულ სტრიქონში, რომელიც შეგეყავს ბრძანებათა სტრიქონიდან ან ვაფორმირებთ პროგრამაში, ხოლო შემდეგ ვანიჭებთ s-ცვლადს. შეიყვანეთ ან დააფორმირეთ რაიმე ახალი სტრიქონი და დააფორმირეთ ახალი სტრიქონი ამ და მოცემული სტრიქონების სიტყვების თანმიმდევრობითი მონაცვლებით, ე. ი პირველი სტრიქონის პირველი სიტყვა, შემდეგ მეორე სტრიქონის მეორე სიტყვა და ა.შ.;

74. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ მეთოდი, რომლითაც განისაზღვრება: $\sqrt{a_1 + \sqrt{a_2 \sqrt{a_3 + \dots + \sqrt{a_n}}}}$;

75. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდის სახით:

$$Y = \begin{cases} h, & x \leq -R-a \\ \frac{h}{a} \times (x-R), & -R-a < x \leq -R \\ \sqrt{R^2 - x^2}, & -R < x \leq R \\ -\sqrt{R^2 - (x-2R)^2}, & R < x \leq 3R \\ \frac{h}{a} (x-3R), & 3R < x \leq 3R+a \\ h, & x > 3R \end{cases}$$

77. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და აღწერეთ თუ რას აკეთებს იგი.

```
using System;

class strigoni
{
    static void Main()
    {
        Console.WriteLine("შეიყვანეთ სრიქონი");
        String sawyisi=Console.ReadLine();
        sawyisi+='\\xff';
        int i=0;
        do
        {
            i++;
        }
        while(sawyisi[i]!='\\xff');
        Console.WriteLine("შედგეი={0}",i);
    }
}
```

78. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და აღწერეთ თუ რას აკეთებს იგი.

```
using System;

namespace jami
{
    class Class1
    {
        static void Main()
        {
            double sum=0;
            double[] a={-2,3,-45,61,8,-9.7,9.9};
            for (int i=0; i<a.Length; i+=2)
            {
                if (a[i]>0)
                    sum+=a[i];
            }
            Console.WriteLine("sum={0}",sum);
        }
    }
}
```

79. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და აღწერეთ თუ რას აკეთებს იგი.

```
using System;
namespace gamosaxuleba
{
    class Class1
    {
        static double meTodi(double x)
        {
            double c=5,h=10, a=6;
            if (x<=0)
                {return 0;}
            else if(x<Math.PI) {return (c+h)/(a*a)*Math.Pow((x-a),2)-h;}
        }
    }
}
```

```

        else {return 0;}
    }
    static void Main()
    {
        Console.WriteLine("{0}", meTodi(2.2));
    }
}

```

80. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და აღწერეთ თუ რას აკეთებს იგი.

```

using System;

class striqoni
{
    static void Main()
    {
        Console.WriteLine("SeiyvaneT sriqoni");
        String sawyisi=Console.ReadLine();
        string Sedegi;
        Sedegi=sawyisi.Replace("nino", "tea ");
        Console.WriteLine("შედეგი={0}", Sedegi);
    }
}

```

81. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და აღწერეთ თუ რას აკეთებს იგი.

```

using System;

namespace jami
{
    class Class1
    {
        static void Main()
        {
            double sum=0;
            double[] a={-4,3,-53,61,8,-9.7,9.9};
            for (int i=1; i<a.Length; i+=2)
            {
                if (a[i]>0)
                    sum+=a[i];
            }
            Console.WriteLine("sum={0}", sum);
        }
    }
}

```

82. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და აღწერეთ თუ რას აკეთებს იგი.

```

using System;
namespace gamosaxuleba
{

```

```

class Class1
{
    static double meTodi(double x)
    {
        double h=10, a=6;
        if (x<=-a)
            {return h;}
        else if(x<0) {return Math.Exp(x)+h;}
        else if(x<a) {return Math.Exp(-x)+h;}
        else {return h;}
    }
    static void Main()
    {
        Console.WriteLine("{0}",meTodi(7.2));
    }
}

```

83. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და აღწერეთ თუ რას აკეთებს იგი.

```

using System;

namespace gamosaxuleba
{
    class Class1
    {
        static double meTodi(double x)
        {
            double c=5,h=10, a=6;
            if (x<=0)
                {return 0;}
            else if(x<=a) {return (c+h)/(a*a)*Math.Pow((x-a),2)-h;}
            else {return 0;}
        }
        static void Main()
        {
            Console.WriteLine("{0}",meTodi(2.2));
        }
    }
}

```

84. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და აღწერეთ თუ რას აკეთებს იგი.

```

using System;

namespace jami
{
    class Class1
    {
        static void Main()
        {int count=0;
        double[] a={-2,3,-45,61,8,-9.7,9.9};
        for (int i=0; i<a.Length; i++)
        {

```

```

        if (a[i]>0)
            count++;
    }
    Console.WriteLine("count={0}", count);
}
}
}

```

85. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და აღწერეთ თუ რას აკეთებს იგი.

```

using System;
namespace qurdaZe
{
    class strigoni
    {
        static void Main()
        {
            Console.WriteLine("SeiyvaneT sriqoni");
            String sawyisi=Console.ReadLine();
            String sededegi="";
            for(int i=0; i<sawyisi.Length; i++)
            {
                if(Char.IsLetter(sawyisi[i]))
                    {sededegi+="X";}
                else
                    {sededegi+=sawyisi[i].ToString();}
            }
            Console.WriteLine("შედეგი={0}", sededegi);
        }
    }
}

```

მეორე ნაწილი

1. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია(+), რომლითაც განისაზღვრება წევრების ჯამი.
2. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ ასოების რაოდენობა.

$$Y = \begin{cases} 0, & x \leq -R \\ \sqrt{R^2 - x^2}, & 0 < x \leq R \\ 0, & x > R \end{cases}$$

3. მოცემულია ორი მატრიცისათვის $A_{m,n}$ და $B_{m,n}$. შეადგინეთ ბინარული ოპერაცია, რომელიც იპოვის C მატრიცას, როგორც ამ მატრიცების წევრობრივ ნამრავლს.
4. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება დადებითი წევრების ჯამი.
5. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს, თუ რამდენჯერ შედის თქვენი სახელი ამ სტრიქონში.

$$Y = \begin{cases} h, & x \leq -a \\ \frac{h}{a^2} \times x^2, & 0 < x \leq a \\ h, & x > a \end{cases}$$

6. მოცემულია ერთნაირგანზომილებიანი მატრიცები A, B . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების ჯამს.

7. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება უარყოფითი წევრების ჯამი.
8. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ 1, თუ თქვენი სახელის მსგავსი თანმიმდევრობა გხვდება ამ სტრიქონში (ანუ სახელის ასოებს შორის შესაძლებელია სხვა სიმბოლოებიც იყოს), ხოლო წინააღმდეგ შემთხვევაში აიღეთ 0.

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{h}{a} \times x, & -a < x \leq 0 \\ a \\ -\frac{h}{a} \times x, & 0 > x \leq a \\ 0, & x > a \end{cases}$$

9. მოცემულია ერთნაირგანზომილებიანი მატრიცები A, B . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების სხვაობას.
10. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება ლუწინდექსებიანი წევრების ჯამი.
11. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ სტრიქონის სიგრძე.

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{c+h}{a^2} \times (x-a)^2 - h, & 0 < x < \pi \\ 0, & x \geq \pi \end{cases}$$

12. მოცემულია ერთნაირგანზომილებიანი მატრიცები A, B . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების ერთმანეთზე განაყოფს.
13. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება კენტინდექსებიანი წევრების ჯამი.
14. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვითვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს, სტრიქონის ბოლოდან რომელ პოზიციაში იწყება თქვენი სახელი (თუკი ასეთი არსებობს, თუ არ არსებობს - 0).

$$Y = \begin{cases} 0, & x \leq -\frac{\pi}{2} \\ \cos x, & -\frac{\pi}{2} < x < \frac{\pi}{2} \\ 0, & x \geq \frac{\pi}{2} \end{cases}$$

15. მოცემულია ერთნაირგანზომილებიანი მატრიცები A, B . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების განაყოფის ნაშთს.
16. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება ლუწინდექსებიანი დადებითი წევრების ჯამი.
17. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვითვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს, რომელი პოზიციიდან შედის თქვენი სახელი ამ სტრიქონში (თუკი ასეთი არსებობს, თუ არ არსებობს - 0).

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{c+h}{a^2} \times (x-a)^2 - h, & 0 < x \leq a \\ 0, & x > a \end{cases}$$

18. მოცემულია ერთნაირგანზომილებიანი მატრიცები A, B. მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც

$$\text{მათი ელემენტების ხარისხს } c_{ij} = a_{ij}^{b_{ij}}.$$

19. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება კენტინდექსებიანი დადებითი წევრების ჯამი.

20. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს, რამდენი "a" ასოა ამ სტრიქონში.

$$Y = \begin{cases} h, & x \leq -a \\ e^x + h, & -a < x < 0 \\ e^{-x} + h, & 0 \leq x < a \\ h, & x \geq a \end{cases}$$

21. მოცემულია ერთნაირგანზომილებიანი მატრიცები A, B. მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების ლოგიკურ ნამრავლს.

22. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება ლუწინდექსებიანი უარყოფითი წევრების ჯამი.
23. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს, რამდენი "b" ასოა ამ სტრიქონში.

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{2h}{a}x, & -0 < x < \frac{a}{2} \\ -\frac{2h}{a}x, & \frac{a}{2} \leq x < a \\ \frac{2h}{a}x, & a \leq x < \frac{3a}{2} \\ 0, & x \geq \frac{3a}{2} \end{cases}$$

24. მოცემულია ერთნაირგანზომილებიანი მატრიცები A, B. მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების ლოგიკურ შეკრებას.
25. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება კენტინდექსებიანი უარყოფითი წევრების ჯამი.
26. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს, რამდენი "c" ასოა ამ სტრიქონში.

$$Y = \begin{cases} 0, & x \leq 0 \\ \operatorname{tg} x, & 0 < x \leq \frac{\pi}{3} \\ h, & \frac{\pi}{3} \leq x \leq a \\ -\frac{h}{b}x, & a < x \leq a + b \\ 0, & x > a + b \end{cases}$$

27. მოცემულია ერთნაირგანზომილებიანი მატრიცები A, B. მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების გამომრიცხავ ლოგიკურ ჯამს(^).
28. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება წევრების რაოდენობა პირველ ნულოვან წევრამდე.
29. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს, რამდენი "d" ასოა ამ სტრიქონში.

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{h}{a^2}x^2, & 0 < x < a \\ \frac{h}{a^2}(x-2a)^2, & a \leq x < 3a \\ \frac{h}{a^2}(x-5a)^2, & 3a \leq x < 5a \\ 0, & x \geq 5a \end{cases}$$

30. მოცემულია ერთნაირგანზომილებიანი მატრიცები A, B . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების კონკატენაციას.
31. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება დადებითი წევრების რაოდენობა.
32. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს, რამდენი "e" ასოა ამ სტრიქონში.

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{c+h}{a^2} \times (x-a)^2 - h, & 0 < x \leq a \\ 0, & x > a \end{cases}$$

33. მოცემულია მატრიცა A . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების ნამრავლს a ცვლადზე.
34. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება უარყოფითი წევრების რაოდენობა.
35. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს, რამდენი სიტყვაა მოცემულ სტრიქონში.

$$Y = \begin{cases} 0, & x \leq -\frac{\pi}{2} \\ \cos x, & -\frac{\pi}{2} < x \leq \frac{\pi}{2} \\ \cos(x - \pi), & \frac{\pi}{2} < x \leq \frac{3}{2}\pi \\ \cos(x - 2\pi), & \frac{3}{2}\pi < x \leq \frac{5}{2}\pi \\ 0, & x > \frac{5}{2}\pi \end{cases}$$

36. მოცემულია მატრიცა A . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების ჯამს a ცვლადზე.
37. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება მაქსიმალური ელემენტი.
38. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს, რამდენი შუალედია სიტყვებს შორის მოცემულ სტრიქონში.

$$Y = \begin{cases} 0, & x \leq -R \\ \sqrt{R^2 - x^2}, & -R < x \leq R \\ 0, & x > R \end{cases}$$

39. მოცემულია მატრიცა A . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების სხვაობას a ცვლადზე.

40. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება მინიმალური ელემენტი მოდულით.

41. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს, რამდენი "X" და "Y" სიმბოლოა მოცემულ სტრიქონში.

$$Y = \begin{cases} 0, & x < a \\ 0, & x \neq n \times a \\ h, & x = n \times a, n = 1, 2, 3, \dots \end{cases}$$

42. მოცემულია მატრიცა A . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების განაყოფს a ცვლადზე.

43. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება მაქსიმალური ელემენტი მოდულით.

44. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს, რამდენი "a" და "b" სიმბოლოა მოცემულ სტრიქონში.

$$Y = \begin{cases} 0, -x \leq a \\ \frac{h}{a} \times (x+a) - h, -a < x \leq 0 \\ \cos x, 0 < x \leq \frac{\pi}{2} \\ \frac{h}{a}(x-\pi), \frac{\pi}{2} < x \leq \pi \\ \cos(x-\pi), \pi < x \leq \frac{3}{2}\pi \\ 0, x > \frac{3}{2}\pi \end{cases}$$

45. მოცემულია მატრიცა A . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების კვადრატების განაყოფს a ცვლადზე.

46. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება კენტი ინდექსებიან ელემენტთა შორის უდიდესი.

47. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს სტრიქონის შუა პოზიციას.

$$Y = \begin{cases} 0, x \leq 0 \\ a, (2n-1)a < x \leq 2na \\ 0, 2na < x \leq (2n+1)a, n = 1, 2, 3, \dots \end{cases}$$

48. მოცემულია მატრიცა A . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების განაყოფის ნაშთს a ცვლადზე.
49. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება ლუწინდექსებიან ელემენტთა შორის უდიდესი.
50. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს მეორე სიტყვის საწყის პოზიციას.

$$Y = \begin{cases} 0, & x \leq 0 \\ \sin x, & 0 < x \leq b \\ e^{-(x-\pi/2)}, & b < x \leq d \\ \frac{\sin(b)}{a} (x-d)x, & d < x \leq d+a \\ 0, & x > d+a \end{cases}$$

51. მოცემულია მატრიცა A . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების ხარისხს a ცვლადით.
52. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება ნულოვანი ელემენტების რაოდენობა.
53. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს ბოლო სიტყვის საწყის პოზიციას.

$$Y = \begin{cases} b, & x \leq -a \\ \frac{h}{a^2} \times x^2 - h, & -a < x \leq a \\ b, & x > a \end{cases}$$

54. მოცემულია მატრიცა A. მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების ლოგიკურ ნამრავლს a ცვლადზე.
55. შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრებ: 1-ის ტოლ ელემენტთა რაოდენობა.
56. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს სტრიქონის გასამმაგებულ სიგრძეს.

$$Y = \begin{cases} 0, & x \leq -a \\ -\frac{h}{a^2} \times x^2 + h, & -a < x \leq a \\ 0, & x > a \end{cases}$$

57. მოცემულია მატრიცა A. მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების ლოგიკურ ჯამს a ცვლადთან.
58. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება წევრების ნამრავლი.
59. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს სტრიქონში შემავალ სიტყვათა რაოდენობას.

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{-h}{a} \times (x+a), & -a < x \leq 0 \\ \frac{h}{a} \times (x-a), & 0 < x \leq a \\ \sin(x-a-\frac{\pi}{2}), & a < x \leq a+\pi \\ 0, & x > a+\pi \end{cases}$$

60. მოცემულია მატრიცა A . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი ელემენტების ურთიერთგამომრიცხავ ლოგიკურ(^) ჯამს a ცვლადთან.
61. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება დადებითი წევრების ნამრავლი.
62. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს სტრიქონში ბოლო სიტყვის საწყის პოზიციას.

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{d-h}{a^2} \times x^2 - d, & -a < x \leq a \\ 0, & x > a \end{cases}$$

63. მოცემულია ორი მატრიციციცისათვის $A_{m,n}$ და $B_{o,p}$ შეადგინეთ ბინარული ოპერაცია, რომელიც იპოვის C მატრიცას, როგორც ამ მატრიცების ნამრავლს. m,n,o,p მატრიცების განზომილებებია შესაბამისად. $n=o$. ისარგებლეთ ფორმულით:

$$C_{i,j} = \sum_{k=1}^n a_{ik} b_{kj}, (i = 1,2,3,\dots, m; j = 1,2,3,\dots, p)$$

64. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება უარყოფითი წევრების ნამრავლი.
65. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს სტრიქონში პირველი სიტყვის ბოლო პოზიციას.

$$Y = \begin{cases} he^{-a}, & x \leq -a \\ he^x, & -a < x \leq 0 \\ he^{-x}, & 0 < x \leq a \\ e^{-a}, & x > a \end{cases}$$

66. მოცემულია მატრიცა A . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი სტრიქონების შესაბამისი ელემენტების ჯამს a ვექტორთან.
67. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება ლუწინდექსიანი ელემენტების ნამრავლი.
68. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს სტრიქონში ბოლო სიტყვის საწყის პოზიციას.

$$Y = \begin{cases} h, & x \leq -a \\ -h, & -a < x \leq a \\ h, & a < x \leq 2a \\ \frac{d}{c}(x-2a), & 2a < x \leq 2a+c \\ 0, & x > 2a+c \end{cases}$$

69. მოცემულია მატრიცა A . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი სტრიქონების შესაბამისი ელემენტების სხვაობას a ვექტორთან.
70. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ უნარული ოპერაცია, რომლითაც განისაზღვრება მოცემული რიცხვების კვადრატების ჯამი.
71. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი. უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს სტრიქონში

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{-h}{a} \times (x+a), & -a < x \leq 0 \\ \frac{h}{a} (x-a), & 0 < x \leq 2a \\ \operatorname{tg}(x-2a), & 2a < x \leq 3a \\ h, & x > 3a \end{cases}$$

72. მოცემულია მატრიცა A . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი სტრიქონების შესაბამისი ელემენტების სხვაობას a ვექტორთან.

73. მოცემულია ნამდვილ რიცხვთა მიმდევრობა a_1, a_2, \dots, a_n , შეადგინეთ

უნარული ოპერაცია, რომლითაც განისაზღვრება: $\sqrt{a_1 + \sqrt{a_2 \sqrt{a_3 + \dots + \sqrt{a_n}}}}$.

74. მოახდინეთ შემდეგი გამოსახულების რეალიზაცია მეთოდების

გადატვირთვით. x შეიძლება იყოს მთელი, ორმაგი ან სტრიქონი.

უკანასკნელის შემთხვევაში არგუმენტის მნიშვნელობად აიღეთ რიცხვი, რომელიც განსაზღვრავს სტრიქონში მეორე სიტყვის საწყის პოზიციას.

$$Y = \begin{cases} h, & x \leq -R - a \\ \frac{h}{a} \times (x - R), & -R - a < x \leq -R \\ \sqrt{R^2 - x^2}, & -R < x \leq R \\ -\sqrt{R^2 - (x - 2R)^2}, & R < x \leq 3R \\ \frac{h}{a} (x - 3R), & 3R < x \leq 3R + a \\ h, & x > 3R \end{cases}$$

75. მოცემულია მატრიცა A . მოახდინეთ ბინარული ოპერაციის რეალიზაცია, რომელიც იპოვის C მატრიცას, როგორც მათი სტრიქონების შესაბამისი ელემენტების განაყოფის ნაშთს a ვექტორთან.

76. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და აღწერეთ თუ რას აკეთებს იგი.

```
using System;
class Matrica
{
    // ეს არის 3X3 მატრიცა
    public static int GANZOMILEBA;
    private bool[,] matrix;
    public Matrica(int GANZ)
    {
        matrix = new bool[GANZ, GANZ];
        GANZOMILEBA=GANZ;
    }

    public Matrica()
    {
        matrix = new bool[3, 3];
        GANZOMILEBA=3;
    }
    // ნებას აძლევს გამომძახებელს მოახდინოს მატრიცის ინიციალიზაცია
    public bool this[int x, int y]
    {
```

```

        get { return matrix[x, y]; }
        set { matrix[x, y] = value; }
    }
    // ნებას აძლევს მომხმარებელს მოახდინოს მატრიცის დამატება
    public static Matrica operator +(Matrica mat1, Matrica mat2)
    {
        Matrica newMatrix = new Matrica();
        for (int x=0; x < GANZOMILEBA; x++)
            for (int y=0; y < GANZOMILEBA; y++)
                newMatrix[x, y] = mat1[x, y] & mat2[x, y];
        return newMatrix;
    }
}
class MatrixTest
{
    //გამოიყენება InitMatrix მეთოდში
    public static Random rand = new Random();
    // Matrica testireba
    static void Main()
    {
        Matrica mat1 = new Matrica();
        Matrica mat2 = new Matrica();
        // მატრიცების შემთხვევითი სიდიდებით ინიციალიზაცია
        InitMatrix(mat1);
        InitMatrix(mat2);
        // მატრიცების ბეჭდვა
        Console.WriteLine("Matrix 1: ");
        MatricisbeWdva(mat1);
        Console.WriteLine("Matrix 2: ");
        MatricisbeWdva(mat2);
        // ოპერაციების შესრულება და შედეგების ბეჭდვა
        Matrica mat3 = mat1 + mat2;
        Console.WriteLine();
        Console.WriteLine("Matrix 1 + Matrix 2 = ");
        MatricisbeWdva(mat3);
    }
    // მატრიცის ელემენტების შევსება შემთხვევითი მნიშვნელობებით
    public static void InitMatrix(Matrica mat)
    {
        for (int x=0; x < Matrica.GANZOMILEBA; x++)
            for (int y=0; y < Matrica.GANZOMILEBA; y++)
                mat[x, y] =
System.Convert.ToBoolean(rand.Next()%2); //NextDouble() შემთხვევითი რიცხვების
გენერირების ფუნქცია
    }
    // მატრიცის ბეჭდვა კონსოლზე
    public static void MatricisbeWdva(Matrica mat)
    {
        Console.WriteLine();
        for (int x=0; x < Matrica.GANZOMILEBA; x++)
        {
            Console.Write("[ ");
            for (int y=0; y < Matrica.GANZOMILEBA; y++)
            {
                // გამოსავალის ფორმატი
                Console.Write("{0,8:#.000000}", mat[x, y]);
                if ((y+1 % 2) < 3)
                    Console.Write(", ");
            }
            Console.WriteLine(" ]");
        }
        Console.WriteLine();
    }
}
}

```

77. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და სიტყვიერად აღწერეთ თუ რას აკეთებს იგი.

```
using System;
namespace gamosaxuleba
{
    class Class1
    {
        static double meTodi(double x)
        {
            double h=10, a=6;
            if (x<=-a)
            {return h;}
            else if(x<0) {return Math.Exp(x)+h;}
            else if(x<a) {return Math.Exp(-x)+h;}
            else {return h;}
        }
        static double meTodi(int x)
        {
            return meTodi((double)x);
        }
        static double meTodi(string x)
        { //აკ იწერება სრიქონიდან რიცხვითი სიდიდის პოვნის ოპერატორები
            return meTodi((double)x.Length);
        }
        static void Main()
        {
            Console.WriteLine("{0}",meTodi(7.2));
            Console.WriteLine("{0}",meTodi(4));
            Console.WriteLine("{0}",meTodi("gela"));
        }
    }
}
```

78. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და აღწერეთ თუ რას აკეთებს იგი.

```
using System;

class sigma// ოპერაციის შემცველი კლასის განსაზღვრა
{
    public static int zoma;
    private double[] V ;
    public void Printp(sigma mat)
    {
        Console.WriteLine();
        for (int x=0; x < sigma.zoma; x++)
        {
            Console.Write("[ ");

            // გამოყვანის ფორმატი
            Console.Write("{0,8:#.000000}", mat[x]);
            Console.WriteLine(" ]");
        }
        Console.WriteLine();
    }
    public static void Printc(sigma mat)
    {
        Console.WriteLine();
        for (int x=0; x < sigma.zoma; x++)
        {
            Console.Write("[ ");
```

```

        // გამოყვანის ფორმატი
        Console.WriteLine("{0,8:#.000000}", mat[x]);
        Console.WriteLine(" ]");
    }
    Console.WriteLine();
}
public sigma(int zomap )//კონსტრუქტორი
{
    V = new double[zomap];// V მასივის ზომის განსაზღვრა
    zoma=zomap;

}
public sigma( )//კონსტრუქტორი
{
    Console.WriteLine("obieqtis zoma gansazRvruli unda iyos ");
}

// ინდექსერი
public double this[int x]
{
    get { return V[x]; }
    set { V[x] = value; }
}

// ოპერაციის განსაზღვრა
public static double operator +(sigma V)
{
    double s=0;
    for (int x=0; x < zoma; x++)
        s+= V[x];
    return s;
}
}

class aTest
{
    // test sigma
    static void Main()
    {
        sigma a = new sigma(7);

        // მატრიცის შევსება
        for (int x=0; x < sigma.zoma; x++)
            a[x] =x+1;
        Console.WriteLine("a ");
        Printa(a);// საწყისი მონაცემების ბეჭდვა
        a.Printp(a);// მეთოდების გამოზახების სხვადასხვა ხერხები
        sigma.Printc(a);
        // ოპერაციის შესრულება და შედეგის ბეჭდვა
        Console.WriteLine("jami ={0} ",+a);
    }

    // a-ს ბეჭდვა კონსოლზე
    public static void Printa(sigma mat)
    {
        Console.WriteLine();
        for (int x=0; x < sigma.zoma; x++)
        {
            Console.Write("[ ");

```

```

        // გამოყვანის ფორმატი
        Console.Write("{0,8:#.000000}", mat[x]);
        Console.WriteLine(" ");
    }
    Console.WriteLine();
}
}

```

79. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და სიტყვიერად აღწერეთ თუ რას აკეთებს იგი.

`using System;`

```

class Matrica
{
    // ეს არის mxn მატრიცა
    public int GANZOMILEBA1;//m
    public int GANZOMILEBA2;//n
    private bool[,] matrix;
    public Matrica(int GANZ1,int GANZ2)
    {
        matrix = new bool[GANZ1, GANZ2];
        GANZOMILEBA1=GANZ1;GANZOMILEBA2=GANZ2;
    }
    public Matrica(int GANZ1)
    {
        matrix = new bool[GANZ1,GANZ1];
        GANZOMILEBA1=GANZ1;GANZOMILEBA2=GANZ1;
    }
    public Matrica()
    {
        matrix = new bool[3, 3];
        GANZOMILEBA1=3;GANZOMILEBA2=3;
    }
    // ნებას აძლევს გამოძახებულს მოახდინოს მატრიცის ინიციალიზაცია
    public bool this[int x, int y]
    {
        get { return matrix[x, y]; }
        set { matrix[x, y] = value; }
    }
    public static Matrica operator &(Matrica mat1, Matrica mat2)
    {
        Matrica axaliMatrica = new
Matrica(mat1.GANZOMILEBA1,mat1.GANZOMILEBA2);
        if(mat1.GANZOMILEBA1==mat2.GANZOMILEBA1 &&
mat1.GANZOMILEBA2==mat2.GANZOMILEBA2)
        {
            for (int x=0; x < mat1.GANZOMILEBA1; x++)
                for (int y=0; y < mat1.GANZOMILEBA2; y++)
                    axaliMatrica[x, y] = mat1[x, y] & mat2[x, y];
        }

        return axaliMatrica;
    }
}
class MatrixTest
{
    //გამოყენება InitMatrix მეთოდში
    public static Random rand = new Random();
    // ტესტირება
    static void Main()
    {
        Matrica mat1 = new Matrica(5,6);
    }
}

```



```

    Matrica mat2 = new Matrica(5,6); //
    InitMatrix(mat1);
    InitMatrix(mat2);
    // matricebis beWdva
    Console.WriteLine("Matrix 1: ");
    MatricisbeWdva(mat1);
    Console.WriteLine("Matrix 2: ");
    MatricisbeWdva(mat2);
    // ოპერაციების შესრულება და შედეგების ბეჭდვა
    Matrica mat3 = mat1 & mat2;
    Console.WriteLine();
    Console.WriteLine("Matrix 1 + Matrix 2 = ");
    MatricisbeWdva(mat3);
}
// მატრიცის ელემენტების შევსება შემთხვევითი მნიშვნელობებით
public static void InitMatrix(Matrica mat)
{
    for (int x=0; x < mat.GANZOMILEBA1; x++)
        for (int y=0; y < mat.GANZOMILEBA2; y++)
            mat[x, y] = System.Convert.ToBoolean(rand.Next()%2);
//Next() SemTxveviti რიცხვების გენერირების ფუნქცია
}

// მატრიცის ბეჭდვა კონსოლზე

public static void MatricisbeWdva(Matrica mat)
{
    Console.WriteLine();
    for (int x=0; x < mat.GANZOMILEBA1; x++)
    {
        Console.Write("[ ");
        for (int y=0; y < mat.GANZOMILEBA2; y++)
        { // გამოსავლის ფორმატი
            Console.Write("{0}", mat[x, y]);
            if ((y+1 % 2) < mat.GANZOMILEBA2)
                Console.Write(", ");
        }
        Console.WriteLine(" ]");
    }
    Console.WriteLine();
}
}

```

80. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და სიტყვიერად აღწერეთ თუ რას აკეთებს იგი.

```

using System;
class Matrica
{ // ეს არის 3x3 მატრიცა
    public int GANZOMILEBA1;
    public int GANZOMILEBA2;
    private double[,] matrix;
    public Matrica(int GANZ1,int GANZ2)
    {
        matrix = new double[GANZ1, GANZ2];
        GANZOMILEBA1=GANZ1;GANZOMILEBA2=GANZ2;
    }
    public Matrica(int GANZ1)
{
    matrix = new double[GANZ1,1];
    GANZOMILEBA1=GANZ1;GANZOMILEBA2=1;
}
    public Matrica()
    {

```

```

        matrix = new double[3, 3];
        GANZOMILEBA1=3;GANZOMILEBA2=3;
    }
    // ნებას აძლევს გამოძახებულს მოახდინოს მატრიცის ინიციალიზაცია
    public double this[int x, int y]
    {
        get { return matrix[x, y]; }
        set { matrix[x, y] = value; }
    }
    // ნებას აძლევს მომხმარებელს მოახდინოს მატრიცის დამატება
    public static Matrica operator +(Matrica mat1, Matrica mat2)
    {Matrica newMatrix = new Matrica(mat1.GANZOMILEBA1,mat1.GANZOMILEBA2);
        if(mat1.GANZOMILEBA1==mat2.GANZOMILEBA1 &&
mat1.GANZOMILEBA2==mat2.GANZOMILEBA2)
        {
            for (int x=0; x < mat1.GANZOMILEBA1; x++)
                for (int y=0; y < mat1.GANZOMILEBA2; y++)
                    newMatrix[x, y] = mat1[x, y] + mat2[x, y];
        }
        else if(mat1.GANZOMILEBA1==mat2.GANZOMILEBA1 &&
mat2.GANZOMILEBA2==1)
        {
            for (int x=0; x < mat1.GANZOMILEBA1; x++)
                for (int y=0; y < mat1.GANZOMILEBA2; y++)
                    newMatrix[x, y] = mat1[x, y] + mat2[x, 0];
        }
        else if(mat1.GANZOMILEBA2==mat2.GANZOMILEBA2 &&
mat2.GANZOMILEBA1==1)
        {
            for (int x=0; x < mat1.GANZOMILEBA1; x++)
                for (int y=0; y < mat1.GANZOMILEBA2; y++)
                    newMatrix[x, y] = mat1[x, y] + mat2[0, y];
        }
        return newMatrix;
    }
}
class MatrixTest
{
    //გამოიყენება Iნიტ ატრის მეთოდში
    public static Random rand = new Random();
    // ტესტირება
    static void Main()
    {
        Matrica mat1 = new Matrica(4,5);
        Matrica mat2 = new Matrica(1,5);//
        // მატრიცების შემთხვევითი სიდიდებით ინიციალიზაცია
        InitMatrix(mat1);
        InitMatrix(mat2);
        // მატრიცების ბეჭდვა
        Console.WriteLine("Matrix 1: ");
        MatricisbeWdva(mat1);
        Console.WriteLine("Matrix 2: ");
        MatricisbeWdva(mat2);
        // ოპერაციების შესრულება და შედეგების ბეჭდვა
        Matrica mat3 = mat1 + mat2;
        Console.WriteLine();
        Console.WriteLine("Matrix 1 + Matrix 2 = ");
        MatricisbeWdva(mat3);
    }
    // მატრიცის ელემენტების შევსება შემთხვევითი მნიშვნელობებით
    public static void InitMatrix(Matrica mat)
    {
        for (int x=0; x < mat.GANZOMILEBA1; x++)

```

```

        for (int y=0; y < mat.GANZOMILEBA2; y++)
            mat[x, y] = rand.NextDouble();
//NextDouble() შემთხვევითი რიცხვების გენერირების მეთოდი(ფუნქცია)
    }
    // მატრიცის ბეჭდვა კონსოლზე
    public static void MatricisbeWdva(Matrica mat)
    {
        Console.WriteLine();
        for (int x=0; x < mat.GANZOMILEBA1; x++)
        {
            Console.Write("[ ");
            for (int y=0; y < mat.GANZOMILEBA2; y++)
            {
                // გამოსავალის ფორმატი
                Console.Write("{0,8:#.000000}", mat[x, y]);
                if ((y+1 % 2) < mat.GANZOMILEBA1)
                    Console.Write(", ");
            }
            Console.WriteLine(" ]");
        }
        Console.WriteLine();
    }
}

```

81. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და სიტყვიერად აღწერეთ თუ რას აკეთებს იგი.

```
using System;
```

```

class Matrica
{
    // ეს არის mxn მატრიცა
    public int GANZOMILEBA1;//m
    public int GANZOMILEBA2;//n
    private bool[,] matrix;
    public Matrica(int GANZ1,int GANZ2)
    {
        matrix = new bool[GANZ1, GANZ2];
        GANZOMILEBA1=GANZ1;GANZOMILEBA2=GANZ2;
    }
    public Matrica(int GANZ1)
    {
        matrix = new bool[GANZ1, GANZ1];
        GANZOMILEBA1=GANZ1;GANZOMILEBA2=GANZ1;
    }
    public Matrica()
    {
        matrix = new bool[3, 3];
        GANZOMILEBA1=3;GANZOMILEBA2=3;
    }
    // ნებას აძლევს გამომძახებელს მოახდინოს მატრიცის ინიციალიზაცია
    public bool this[int x, int y]
    {
        get { return matrix[x, y]; }
        set { matrix[x, y] = value; }
    }
    public static Matrica operator |(Matrica mat1, Matrica mat2)
    {
        Matrica axaliMatrica = new
Matrica(mat1.GANZOMILEBA1,mat1.GANZOMILEBA2);
        if(mat1.GANZOMILEBA1==mat2.GANZOMILEBA1 &&
mat1.GANZOMILEBA2==mat2.GANZOMILEBA2)
        {
            for (int x=0; x < mat1.GANZOMILEBA1; x++)
                for (int y=0; y < mat1.GANZOMILEBA2; y++)

```

```

        axaliMatrica[x, y] = mat1[x, y] | mat2[x, y];
    }

    return axaliMatrica;
}
}
class MatrixTest
{
    //გამოიყენება InitMatrix მეთოდში
    public static Random rand = new Random();
    // ტესტირება
    static void Main()
    {
        Matrica mat1 = new Matrica(5,6);
        Matrica mat2 = new Matrica(5,6);//
        InitMatrix(mat1);
        InitMatrix(mat2);
        // მატრიცების ბეჭდვა
        Console.WriteLine("Matrix 1: ");
        MatricisbeWdva(mat1);
        Console.WriteLine("Matrix 2: ");
        MatricisbeWdva(mat2);
        // ოპერაციების შესრულება და შედეგების ბეჭდვა
        Matrica mat3 = mat1 | mat2;
        Console.WriteLine();
        Console.WriteLine("Matrix 1 + Matrix 2 =");
        MatricisbeWdva(mat3);
    }
    // მატრიცის ელემენტების შევსება შემთხვევითი მნიშვნელობებით
    public static void InitMatrix(Matrica mat)
    {
        for (int x=0; x < mat.GANZOMILEBA1; x++)
            for (int y=0; y < mat.GANZOMILEBA2; y++)
                mat[x, y] = System.Convert.ToBoolean(rand.Next()%2);
    }
    //Next() შემთხვევითი რიცხვების გენერირების მეთოდი(ფუნქცია)
}

// მატრიცის ბეჭდვა კონსოლზე

public static void MatricisbeWdva(Matrica mat)
{
    Console.WriteLine();
    for (int x=0; x < mat.GANZOMILEBA1; x++)
    {
        Console.Write("[ ");
        for (int y=0; y < mat.GANZOMILEBA2; y++)
        {
            // გამოსავალის ფორმატი
            Console.Write("{0}", mat[x, y]);
            if ((y+1 % 2) < mat.GANZOMILEBA2)
                Console.Write(", ");
        }
        Console.WriteLine(" ]");
    }
    Console.WriteLine();
}
}

```

82. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და სიტყვიერად აღწერეთ თუ რას აკეთებს იგი

```

using System;

class Matrica
{
    private double[,] matrix;
    private static int DIMSIZEp;

```

```

// ეს ორგანზომილებიანი მატრიცაა
public Matrica(int DIMSIZE)
{
matrix = new double[DIMSIZE, DIMSIZE];
    DIMSIZEp=DIMSIZE;
}

// მატრიცის ინიციალიზაციის უფლებას აძლევს გამოძახებულს
public double this[int x, int y]
{
    get { return matrix[x, y]; }
    set { matrix[x, y] = value; }
}

// მომხმარებელს აძლევს მატრიცების შეკრების საშუალებას
public static Matrica operator +(Matrica mat1, Matrica mat2)
{
    Matrica newMatrix = new Matrica(DIMSIZEp);

    for (int x=0; x < DIMSIZEp; x++)
        for (int y=0; y < DIMSIZEp; y++)
            newMatrix[x, y] = mat1[x, y] + mat2[x, y];

    return newMatrix;
}
public static void InitMatrix(Matrica mat)
{
    Random rand = new Random();
    for (int x=0; x < DIMSIZEp; x++)
        for (int y=0; y < DIMSIZEp; y++)
            mat[x, y] = rand.NextDouble();
}
public static void PrintMatrix(Matrica mat)
{
    Console.WriteLine();
    for (int x=0; x < DIMSIZEp; x++)
    {
        Console.Write("[ ");
        for (int y=0; y < DIMSIZEp; y++)
        {
            Console.Write("{0,8:#.000000}", mat[x, y]);

            if ((y+1%2) < DIMSIZEp)
                Console.Write(", ");
        }
        Console.WriteLine(" ]");
    }
    Console.WriteLine();
}

class MatrixTest
{
    static void Main()
    {
        Matrica mat1 = new Matrica(7);
        Matrica mat2 = new Matrica(7);

        // მატრიცის ელემენტებისადმი შემთხვევითი მნიშვნელობების მინიჭება
        Matrica.InitMatrix(mat1);
        Matrica.InitMatrix(mat2);

        // საწყისი მატრიცების ბეჭდვა
        Console.WriteLine("Matrix 1: ");
        Matrica.PrintMatrix(mat1);
        Console.WriteLine("Matrix 2: ");
    }
}

```

```

Matrica.PrintMatrix(mat2);

// ოპერაციის შესრულება და მატრიცების ბეჭდვა
Matrica mat3 = mat1 + mat2;

Console.WriteLine();
Console.WriteLine("Matrix 1 + Matrix 2 = ");
Matrica.PrintMatrix(mat3);
}

}

```

83. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და სიტყვიერად აღწერეთ თუ რას აკეთებს იგი

```

using System;

class Matrica
{
    // ეს არის mxn მატრიცა
    public int GANZOMILEBA1;//m
    public int GANZOMILEBA2;//n
    private double[,] matrix;
    public Matrica(int GANZ1,int GANZ2)
    {
        matrix = new double[GANZ1, GANZ2];
        GANZOMILEBA1=GANZ1;GANZOMILEBA2=GANZ2;
    }
    public Matrica(int GANZ1)
    {
        matrix = new double[GANZ1,GANZ1];
        GANZOMILEBA1=GANZ1;GANZOMILEBA2=GANZ1;
    }
    public Matrica()
    {
        matrix = new double[3, 3];
        GANZOMILEBA1=3;GANZOMILEBA2=3;
    }
    // ნებას აძლევს გამოძახებულს მოახდინოს მატრიცის ინიციალიზაცია
    public double this[int x, int y]
    {
        get { return matrix[x, y]; }
        set { matrix[x, y] = value; }
    }
    public static Matrica operator +(Matrica mat1, Matrica mat2)
    {
        Matrica axaliMatrica = new
Matrica(mat1.GANZOMILEBA1,mat1.GANZOMILEBA2);
        if(mat1.GANZOMILEBA1==mat2.GANZOMILEBA1 &&
mat1.GANZOMILEBA2==mat2.GANZOMILEBA2)
        {
            for (int x=0; x < mat1.GANZOMILEBA1; x++)
                for (int y=0; y < mat1.GANZOMILEBA2; y++)
                    axaliMatrica[x, y] = mat1[x, y] + mat2[x, y];
        }

        return axaliMatrica;
    }
}
class MatrixTest
{
    //გამოიყენება InitMatrix მეთოდში

```

```

public static Random rand = new Random();
// ტესტირება
static void Main()
{
    Matrica mat1 = new Matrica(5,6);
    Matrica mat2 = new Matrica(5,6);
    InitMatrix(mat1);
    InitMatrix(mat2);
    // მატრიცების ბეჭდვა
    Console.WriteLine("Matrix 1: ");
    MatricisbeWdva(mat1);
    Console.WriteLine("Matrix 2: ");
    MatricisbeWdva(mat2);
    // ოპერაციების შესრულება და შედეგების ბეჭდვა
    Matrica mat3 = mat1 + mat2;
    Console.WriteLine();
    Console.WriteLine("Matrix 1 + Matrix 2 = ");
    MatricisbeWdva(mat3);
}
// მატრიცის ელემენტების შევსება შემთხვევითი მნიშვნელობებით
public static void InitMatrix(Matrica mat)
{
    for (int x=0; x < mat.GANZOMILEBA1; x++)
        for (int y=0; y < mat.GANZOMILEBA2; y++)
            mat[x, y] = rand.NextDouble();
}
//Next() შემთხვევითი რიცხვების გენერირების ფუნქცია
}

// მატრიცის ბეჭდვა კონსოლზე

public static void MatricisbeWdva(Matrica mat)
{
    Console.WriteLine();
    for (int x=0; x < mat.GANZOMILEBA1; x++)
    {
        Console.Write("[ ");
        for (int y=0; y < mat.GANZOMILEBA2; y++)
        { // გამოსავალის ფორმატი
            Console.Write("{0,8:#.000000}", mat[x, y]);
            if ((y+1 % 2) < mat.GANZOMILEBA2)
                Console.Write(", ");
        }
        Console.WriteLine(" ]");
    }
    Console.WriteLine();
}
}

```

84. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და სიტყვიერად აღწერეთ თუ რას აკეთებს იგი

```

using System;
class Vectx
{
    public int GANZOMILEBAv;
    private bool[] vectx;
    public Vectx(int GANZ1)
    {
        vectx = new bool[GANZ1];
        GANZOMILEBAv=GANZ1;
    }
    public bool this[int x]
    {
        get { return vectx[x]; }
        set { vectx[x] = value; }
    }
}

```

```

    }
}
class Matrica
{
    // ეს არის mxn მატრიცა
    public int GANZOMILEBA1;
    public int GANZOMILEBA2;
    private bool[,] matrix;
    public Matrica(int GANZ1,int GANZ2)
    {
        matrix = new bool[GANZ1, GANZ2];
        GANZOMILEBA1=GANZ1;GANZOMILEBA2=GANZ2;
    }
    public Matrica(int GANZ1)
    {
        matrix = new bool[GANZ1,GANZ1];
        GANZOMILEBA1=GANZ1;GANZOMILEBA2=GANZ1;
    }
    public Matrica()
    {
        matrix = new bool[3, 3];
        GANZOMILEBA1=3;GANZOMILEBA2=3;
    }
    // ნებას აძლევს გამოძახებულს მოახდინოს მატრიცის ინიციალიზაცია
    public bool this[int x, int y]
    {
        get { return matrix[x, y]; }
        set { matrix[x, y] = value; }
    }

    public static Matrica operator &(Matrica mat1, Vectx vectori)
    {
        Matrica newMatrix = new
Matrica(mat1.GANZOMILEBA1,mat1.GANZOMILEBA2);

        if(mat1.GANZOMILEBA1==vectori.GANZOMILEBAv)
        {
            for (int x=0; x < mat1.GANZOMILEBA1; x++)
                for (int y=0; y < mat1.GANZOMILEBA2; y++)
                    newMatrix[x, y] = mat1[x, y] & vectori[x];
        }
        else if(mat1.GANZOMILEBA2==vectori.GANZOMILEBAv)
        {
            for (int x=0; x < mat1.GANZOMILEBA1; x++)
                for (int y=0; y < mat1.GANZOMILEBA2; y++)
                    newMatrix[x, y] = mat1[x, y] & vectori[y];
        }
        return newMatrix;
    }
}
class MatrixTest
{
    //გამოყენება InitMatrix მეთოდში
    public static Random rand = new Random();
    // ტესტირება
    static void Main()
    {
        Matrica mat1 = new Matrica(5,6);
        Vectx vectori = new Vectx(6);//
        InitMatrix(mat1);
        InitMatrix(vectori);
        // მატრიცების ბეჭდვა
        Console.WriteLine("Matrix 1: ");
        MatricisbeWdva(mat1);
        Console.WriteLine("Vectx 2: ");
        MatricisbeWdva(vectori);
    }
}

```



```

// ოპერაციების შესრულება და შედეგების ბეჭდვა
Matrica mat3 = mat1 & vector1;
Console.WriteLine();
Console.WriteLine("Matrix 1 + Vectx 2 = ");
MatricisbeWdva(mat3);
}
// მატრიცის ელემენტების შევსება შემთხვევითი მნიშვნელობებით
public static void InitMatrix(Matrica mat)
{
    for (int x=0; x < mat.GANZOMILEBA1; x++)
        for (int y=0; y < mat.GANZOMILEBA2; y++)
            mat[x, y] = Convert.ToBoolean(rand.Next()%2);
//Nextbool() შემთხვევითი რიცხვების გენერირების ფუნქცია
}
public static void InitMatrix(Vectx mat)
{
    for (int x=0; x < mat.GANZOMILEBAv; x++)
        mat[x] = Convert.ToBoolean(rand.Next()%2);
//Nextbool() შემთხვევითი რიცხვების გენერირების მეთოდი(ფუნქცია)
}
// მატრიცის ბეჭდვა კონსოლზე
public static void MatricisbeWdva(Vectx mat)
{
    Console.WriteLine();int y=0;Console.Write("[ ");
    for (int x=0; x < mat.GANZOMILEBAv; x++)
    {

        Console.Write("{0,8:#.000000}", mat[x]);
        if ((y+1 % 2) < mat.GANZOMILEBAv)
            {Console.Write(", ");y++;}
    }
    Console.WriteLine(" ]");

    Console.WriteLine();
}
public static void MatricisbeWdva(Matrica mat)
{
    Console.WriteLine();
    for (int x=0; x < mat.GANZOMILEBA1; x++)
    {
        Console.Write("[ ");
        for (int y=0; y < mat.GANZOMILEBA2; y++)
        { // გამოსავლის ფორმატი
            Console.Write("{0,8:#.000000}", mat[x, y]);
            if ((y+1 % 2) < mat.GANZOMILEBA2)
                Console.Write(", ");
        }
        Console.WriteLine(" ]");
    }
    Console.WriteLine();
}
}
}

```

85. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და სიტყვიერად აღწერეთ თუ რას აკეთებს იგი.

```

using System;
class Vectx
{
    public int GANZOMILEBAv;
    private bool[] vectx;
    public Vectx(int GANZ1)
    {
        vectx = new bool[GANZ1];
    }
}

```

```

        GANZOMILEBAv=GANZ1;
    }
    public bool this[int x]
    {
        get { return vectx[x]; }
        set { vectx[x] = value; }
    }
}
class Matrica
{
    // ეს არის mxn მატრიცა
    public int GANZOMILEBA1;
    public int GANZOMILEBA2;
    private bool[,] matrix;
    public Matrica(int GANZ1,int GANZ2)
    {
        matrix = new bool[GANZ1, GANZ2];
        GANZOMILEBA1=GANZ1;GANZOMILEBA2=GANZ2;
    }
    public Matrica(int GANZ1)
    {
        matrix = new bool[GANZ1,GANZ1];
        GANZOMILEBA1=GANZ1;GANZOMILEBA2=GANZ1;
    }
    public Matrica()
    {
        matrix = new bool[3, 3];
        GANZOMILEBA1=3;GANZOMILEBA2=3;
    }
    // ნებას აძლევს გამომძახებულს მოახდინოს მატრიცის ინიციალიზაცია
    public bool this[int x, int y]
    {
        get { return matrix[x, y]; }
        set { matrix[x, y] = value; }
    }

    public static Matrica operator |(Matrica mat1, Vectx vectori)
    {
        Matrica newMatrix = new
Matrica(mat1.GANZOMILEBA1,mat1.GANZOMILEBA2);

        if(mat1.GANZOMILEBA1==vectori.GANZOMILEBAv)
        {
            for (int x=0; x < mat1.GANZOMILEBA1; x++)
                for (int y=0; y < mat1.GANZOMILEBA2; y++)
                    newMatrix[x, y] = mat1[x, y] & vectori[x];
        }
        else if(mat1.GANZOMILEBA2==vectori.GANZOMILEBAv)
        {
            for (int x=0; x < mat1.GANZOMILEBA1; x++)
                for (int y=0; y < mat1.GANZOMILEBA2; y++)
                    newMatrix[x, y] = mat1[x, y] | vectori[y];
        }
        return newMatrix;
    }
}
class MatrixTest
{
    //გამოყენება InitMatrix მეთოდში
    public static Random rand = new Random();
    // ტესტირება
    static void Main()
    {
        Matrica mat1 = new Matrica(5,6);
        Vectx vectori = new Vectx(6);//
        InitMatrix(mat1);
    }
}

```

```

    InitMatrix(vectori);
    // მატრიცების ბეჭდვა
    Console.WriteLine("Matrix 1: ");
    MatricisbeWdva(mat1);
    Console.WriteLine("Vectx 2: ");
    MatricisbeWdva(vectori);
    // ოპერაციების შესრულება და შედეგების ბეჭდვა
    Matrica mat3 = mat1 | vectori;
    Console.WriteLine();
    Console.WriteLine("Matrix 1 + Vectx 2 = ");
    MatricisbeWdva(mat3);
}
// მატრიცის ელემენტების შევსება შემთხვევითი მნიშვნელობებით
public static void InitMatrix(Matrica mat)
{
    for (int x=0; x < mat.GANZOMILEBA1; x++)
        for (int y=0; y < mat.GANZOMILEBA2; y++)
            mat[x, y] = Convert.ToBoolean(rand.Next()%2);
//Nextbool() SemTxveviTi ricxvebis generirebis funqcia
}
public static void InitMatrix(Vectx mat)
{
    for (int x=0; x < mat.GANZOMILEBAv; x++)
        mat[x] = Convert.ToBoolean(rand.Next()%2);
//Nextbool() შემთხვევითი რიცხვების გენერირების მეთოდი(ფუნქცია)
}
// მატრიცის ბეჭდვა კონსოლზე
public static void MatricisbeWdva(Vectx mat)
{
    Console.WriteLine();int y=0;Console.Write("[ ");
    for (int x=0; x < mat.GANZOMILEBAv; x++)
    {
        Console.Write("{0,8:#.000000}", mat[x]);
        if ((y+1 % 2) < mat.GANZOMILEBAv)
            {Console.Write(", ");y++;}
    }
    Console.WriteLine(" ]");

    Console.WriteLine();
}
public static void MatricisbeWdva(Matrica mat)
{
    Console.WriteLine();
    for (int x=0; x < mat.GANZOMILEBA1; x++)
    {
        Console.Write("[ ");
        for (int y=0; y < mat.GANZOMILEBA2; y++)
        { // გამოსავლის ფორმატი
            Console.Write("{0,8:#.000000}", mat[x, y]);
            if ((y+1 % 2) < mat.GANZOMILEBA2)
                Console.Write(", ");
        }
        Console.WriteLine(" ]");
    }
    Console.WriteLine();
}
}
}

```

86. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და სიტყვიერად აღწერეთ თუ რას აკეთებს იგი.

```

using System;
class Vectx

```

```

{
public int GANZOMILEBAv;
private double[] vectx;
public Vectx(int GANZ1)
{
    vectx = new double[GANZ1];
    GANZOMILEBAv=GANZ1;
}
public double this[int x]
{
    get { return vectx[x]; }
    set { vectx[x] = value; }
}
}
class Matrica
{
    // ეს არის mxn მატრიცა
    public int GANZOMILEBA1;
    public int GANZOMILEBA2;
    private double[,] matrix;
    public Matrica(int GANZ1,int GANZ2)
    {
        matrix = new double[GANZ1, GANZ2];
        GANZOMILEBA1=GANZ1;GANZOMILEBA2=GANZ2;
    }
    public Matrica(int GANZ1)
{
    matrix = new double[GANZ1,GANZ1];
    GANZOMILEBA1=GANZ1;GANZOMILEBA2=GANZ1;
}
    public Matrica()
    {
        matrix = new double[3, 3];
        GANZOMILEBA1=3;GANZOMILEBA2=3;
    }
    // ნებას აძლევს გამოძახებულს მოახდინოს მატრიცის ინიციალიზაცია
    public double this[int x, int y]
    {
        get { return matrix[x, y]; }
        set { matrix[x, y] = value; }
    }

    public static Matrica operator +(Matrica mat1, Vectx vectori)
    {Matrica newMatrix = new Matrica(mat1.GANZOMILEBA1,mat1.GANZOMILEBA2);

        if(mat1.GANZOMILEBA1==vectori.GANZOMILEBAv)
        {
            for (int x=0; x < mat1.GANZOMILEBA1; x++)
                for (int y=0; y < mat1.GANZOMILEBA2; y++)
                    newMatrix[x, y] = mat1[x, y] + vectori[x];
        }
        else if(mat1.GANZOMILEBA2==vectori.GANZOMILEBAv)
        {
            for (int x=0; x < mat1.GANZOMILEBA1; x++)
                for (int y=0; y < mat1.GANZOMILEBA2; y++)
                    newMatrix[x, y] = mat1[x, y] + vectori[y];
        }
        return newMatrix;
    }
}
class MatrixTest
{
    //გამოყენება InitMatrix მეთოდში
    public static Random rand = new Random();
    // ტესტირება
    static void Main()

```

```

{
    Matrica mat1 = new Matrica(5,6);
    Vectx vectori = new Vectx(6);//
    InitMatrix(mat1);
    InitMatrix(vectori);
    // მატრიცების ბეჭდვა
    Console.WriteLine("Matrix 1: ");
    MatricisbeWdva(mat1);
    Console.WriteLine("Vectx 2: ");
    MatricisbeWdva(vectori);
    // ოპერაციების შესრულება და შედეგების ბეჭდვა
    Matrica mat3 = mat1 + vectori;
    Console.WriteLine();
    Console.WriteLine("Matrix 1 + Vectx 2 = ");
    MatricisbeWdva(mat3);
}
// მატრიცის ელემენტების შევსება შემთხვევითი მნიშვნელობებით
public static void InitMatrix(Matrica mat)
{
    for (int x=0; x < mat.GANZOMILEBA1; x++)
        for (int y=0; y < mat.GANZOMILEBA2; y++)
            mat[x, y] = rand.NextDouble();
//NextDouble() შემთხვევითი რიცხვების გენერირების მეთოდი(ფუნქცია)
}
public static void InitMatrix(Vectx mat)
{
    for (int x=0; x < mat.GANZOMILEBAv; x++)
        mat[x] = rand.NextDouble();
//NextDouble() შემთხვევითი რიცხვების გენერირების მეთოდი(ფუნქცია)
}
// მატრიცის ბეჭდვა კონსოლზე
public static void MatricisbeWdva(Vectx mat)
{
    Console.WriteLine();int y=0;Console.Write("[ ");
    for (int x=0; x < mat.GANZOMILEBAv; x++)
    {

        Console.Write("{0,8:#.000000}", mat[x]);
        if ((y+1 % 2) < mat.GANZOMILEBAv)
            {Console.Write(", ");y++;}
        }
        Console.WriteLine(" ]");

    Console.WriteLine();
}
public static void MatricisbeWdva(Matrica mat)
{
    Console.WriteLine();
    for (int x=0; x < mat.GANZOMILEBA1; x++)
    {
        Console.Write("[ ");
        for (int y=0; y < mat.GANZOMILEBA2; y++)
        {
            // გამოსავლის ფორმატი
            Console.Write("{0,8:#.000000}", mat[x, y]);
            if ((y+1 % 2) < mat.GANZOMILEBA1)
                Console.Write(", ");
        }
        Console.WriteLine(" ]");
    }
    Console.WriteLine();
}
}

```

87. მოახდინეთ მოცემული პროგრამის ვერიფიკაცია(ტექსტის ანალიზი) და სიტყვიერად აღწერეთ თუ რას აკეთებს იგი.

```
using System;

class Matrix3D
{
    // ეს არის ორგანზომილებიანი მატრიცა
    public const int DIMSIZE = 3;
    private double[,] matrix = new double[DIMSIZE, DIMSIZE];

    // უფლებას აძლევს გამოძახებულს შეასრულოს მატრიცის ინიციალიზაცია
    public double this[int x, int y]
    {
        get { return matrix[x, y]; }
        set { matrix[x, y] = value; }
    }

    // მატრიცების შეკრების უფლებას აძლევს გამოძახებულს
    public static Matrix3D operator +(Matrix3D mat1, Matrix3D mat2)
    {
        Matrix3D newMatrix = new Matrix3D();

        for (int x=0; x < DIMSIZE; x++)
            for (int y=0; y < DIMSIZE; y++)
                newMatrix[x, y] = mat1[x, y] + mat2[x, y];

        return newMatrix;
    }
}

class MatrixTest
{
    // გამოიყენება InitMatrix მეთოდში.
    public static Random rand = new Random();

    // ტესტირება Matrix3D-ის
    static void Main()
    {
        Matrix3D mat1 = new Matrix3D();
        Matrix3D mat2 = new Matrix3D();

        // მატრიცების შევსება შემთხვევითი სიდიდებით
        InitMatrix(mat1);
        InitMatrix(mat2);

        // მატრიცების ბეჭდვა
        Console.WriteLine("Matrix 1: ");
        PrintMatrix(mat1);
        Console.WriteLine("Matrix 2: ");
        PrintMatrix(mat2);

        // ოპერაციის შესრულება და მატრიცის ბეჭდვა
        Matrix3D mat3 = mat1 + mat2;

        Console.WriteLine();
        Console.WriteLine("Matrix 1 + Matrix 2 = ");
        PrintMatrix(mat3);
    }

    // მატრიცის შევსება შემთხვევითი სიდიდებით
    public static void InitMatrix(Matrix3D mat)
    {
        for (int x=0; x < Matrix3D.DIMSIZE; x++)
```

```

        for (int y=0; y < Matrix3D.DIMSIZE; y++)
            mat[x, y] = rand.NextDouble();
    }

    // მატრიცის ბეჭდვა კონსოლზე
    public static void PrintMatrix(Matrix3D mat)
    {
        Console.WriteLine();
        for (int x=0; x < Matrix3D.DIMSIZE; x++)
        {
            Console.Write("[ ");
            for (int y=0; y < Matrix3D.DIMSIZE; y++)
            {
                // გამოყვანის ფორმატი
                Console.Write("{0,8:#.000000}", mat[x, y]);

                if ((y+1 % 2) < 3)
                    Console.Write(", ");
            }
            Console.WriteLine(" ]");
        }
        Console.WriteLine();
    }
}

```

მესამე ნაწილი

1. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox-ში მოცემული სტრიქონიდან იპოვის თქვენს სახელში შემავალი ასოების რაოდენობას(თუ ასეთი არსებობს) და შედეგს აჩვენებს მეორე TextBox-ში;
2. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს წევრების ჯამს და შედეგს გამოიტანს TextBox-ში;
3. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq -R \\ \sqrt{R^2 - x^2}, & 0 < x \leq R \\ 0, & x > R \end{cases}$$

4. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox1-ში მოცემული სტრიქონიდან იპოვის, თუ რამდენჯერ შედის თქვენი სახელი ამ სტრიქონში და შედეგს გამოიტანს TextBox2-ში;
5. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს დადებითი წევრების ჯამს და შედეგს გამოიტანს TextBox-ში;
6. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} h, & x \leq -a \\ \frac{h}{a^2} \times x^2, & 0 < x \leq a \\ h, & x > a \end{cases}$$

7. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox1-ში მოცემული სტრიქონიდან დაადგენს არსებობს თუ არა თქვენი სახელის მსგავსი თანმიმდევრობა ამ სტრიქონში (ანუ სახელის ასოებს შორის შესაძლებელია სხვა სიმბოლოებიც იყოს) და შედეგს გამოიტანს TextBox2-ში;
8. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს უარყოფითი წევრების ჯამს და შედეგს გამოიტანს TextBox-ში ;
9. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{h}{-a} \times x, & -a < x \leq 0 \\ a \\ \frac{h}{a} \times x, & 0 > x \leq a \\ 0, & x > a \end{cases}$$

10. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox1-ში მოცემული სტრიქონიდან იპოვის სტრიქონის სიგრძეს ფუნქციის გამოყენების გარეშე და შედეგს გამოიტანს TextBox2-ში ;
11. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს ლუწინდექსებიანი წევრების ჯამს და შედეგს გამოიტანს TextBox-ში;
12. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{c+h}{a^2} \times (x-a)^2 - h, & 0 < x < \pi \\ 0, & x \geq \pi \end{cases}$$

13. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox1-ში მოცემული სტრიქონიდან იპოვის სტრიქონის ბოლოდან რომელ პოზიციაშია თქვენი სახელი (თუკი ასეთი არსებობს) და შედეგს გამოიტანს TextBox2-ში;
14. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს კენტინდექსებიანი წევრების ჯამს და შედეგს გამოიტანს TextBox-ში;
15. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს, ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq -\frac{\pi}{2} \\ \cos x, & -\frac{\pi}{2} < x < \frac{\pi}{2} \\ 0, & x \geq \frac{\pi}{2} \end{cases}$$

16. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox1-ში მოცემული სტრიქონიდან იპოვის რომელი პოზიციებიდან შედის თქვენი სახელი ამ სტრიქონში და შედეგს გამოიტანს TextBox2-ში;
17. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს ლუწინდექსებიანი დადებითი წევრების ჯამს და შედეგს გამოიტანს TextBox-ში;

18. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წვერების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{c+h}{a^2} \times (x-a)^2 - h, & 0 < x \leq a \\ 0, & x > a \end{cases}$$

19. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox1-ში მოცემულ სტრიქონში შეცვლის ყოველ თქვენ სახელს რომელიმე, მაგალითად, თქვენი მეგობრის სახელით და შედეგს გამოიტანს TextBox2-ში ;
20. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს კენტინდექსებიანი დადებითი წვერების ჯამს და შედეგს გამოიტანს TextBox-ში;
21. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წვერების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} h, & x \leq -a \\ e^x + h, & -a < x < 0 \\ e^{-x} + h, & 0 \leq x < a \\ h, & x \geq a \end{cases}$$

22. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox1-ში მოცემულ სტრიქონში დააღაგებს სიტყვებს უკუმიმდევრობით და შედეგს გამოიტანს TextBox2-ში;
23. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს ლუწინდექსებიანი უარყოფითი წვერების ჯამს და შედეგს გამოიტანს TextBox-ში;

24. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წვერების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, x \leq 0 \\ \frac{2h}{a}x, -0 < x < \frac{a}{2} \\ -\frac{2h}{a}x, \frac{a}{2} \leq x < a \\ \frac{2h}{a}x, a \leq x < \frac{3a}{2} \\ 0, x \geq \frac{3a}{2} \end{cases}$$

25. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox1-ში მოცემულ სტრიქონში დააღაგებს ასოებს უკუმიმდევრობით და შედეგს გამოიტანს TextBox2-ში;
26. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს კენტინდექსებიანი უარყოფითი წვერების ჯამს და შედეგს გამოიტანს TextBox-ში;
27. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წვერების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq 0 \\ \operatorname{tg} x, & 0 < x \leq \frac{\pi}{3} \\ h, & \frac{\pi}{3} \leq x \leq a \\ -\frac{h}{b}x, & a < x \leq a + b \\ 0, & x > a + b \end{cases}$$

28. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox1-ში მოცემულ სტრიქონში დააღებებს სიტყვებს ანბანის მიხედვით და შედეგს გამოიტანს TextBox2-ში;
29. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს წევრების რაოდენობა პირველ ნულოვან წევრამდე და შედეგს გამოიტანს TextBox-ში;
30. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს, ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{h}{a^2}x^2, & 0 < x < a \\ \frac{h}{a^2}(x-2a)^2, & a \leq x < 3a \\ \frac{h}{a^2}(x-5a)^2, & 3a \leq x < 5a \\ 0, & x \geq 5a \end{cases}$$

31. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox1-ში მოცემულ სტრიქონში შეცვალის ყოველ ასოს X-ით და შედეგს გამოიტანს TextBox2-ში;
32. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს დადებითი წევრების რაოდენობას და შედეგს გამოიტანს TextBox-ში;
33. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq 0 \\ \frac{c+h}{a^2} \times (x-a)^2 - h, & 0 < x \leq a \\ 0, & x > a \end{cases}$$

34. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox1-ში მოცემულ სტრიქონში სიტყვებს შორის დატოვებს მხოლოდ ერთი შუალედის სიმბოლოს და შედეგს გამოიტანს TextBox2-ში;
35. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს უარყოფითი წევრების რაოდენობას და შედეგს გამოიტანს TextBox-ში;
36. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq -\frac{\pi}{2} \\ \cos x, & -\frac{\pi}{2} < x \leq \frac{\pi}{2} \\ \cos(x - \pi), & \frac{\pi}{2} < x \leq \frac{3}{2}\pi \\ \cos(x - 2\pi), & \frac{3}{2}\pi < x \leq \frac{5}{2}\pi \\ 0, & x > \frac{5}{2}\pi \end{cases}$$

37. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox1-ში მოცემულ სტრიქონში სიტყვებს შორის შუალედი შეცვალეთ სიმბოლო Z-ით და შედეგს გამოიტანს TextBox2-ში;
38. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს მაქსიმალური ელემენტს და შედეგს გამოიტანს TextBox-ში;
39. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს, ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq -R \\ \sqrt{R^2 - x^2}, & -R < x \leq R \\ 0, & x > R \end{cases}$$

40. შექმენით WINDOWS აპლიკაცია, რომელიც პირველ TextBox1-ში მოცემულ სტრიქონში. თითოეული სიტყვა მოაქციეთ X და Z სიმბოლოებს შორის და შედეგს გამოიტანს TextBox2-ში;
41. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს მინიმალური ელემენტს მოდულით და შედეგს გამოიტანს TextBox-ში;

42. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, x < a \\ 0, x \neq n \times a \\ h, x = n \times a, n = 1, 2, 3, \dots \end{cases}$$

43. შექმენით WINDOWS აპლიკაცია, რომელიც TextBox1-ში მოცემულ სტრიქონს მიუწერს TextBox2-ში მოცემულ სტრიქონს მარცხნიდან და შედეგს გამოიტანს TextBox3-ში;
44. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს მაქსიმალური ელემენტს მოდულით და შედეგს გამოიტანს TextBox-ში;
45. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, -x \leq a \\ \frac{h}{a} \times (x + a) - h, -a < x \leq 0 \\ \cos x, 0 < x \leq \frac{\pi}{2} \\ \frac{h}{a} (x - \pi), \frac{\pi}{2} < x \leq \pi \\ \cos(x - \pi), \pi < x \leq \frac{3}{2} \pi \\ 0, x > \frac{3}{2} \pi \end{cases}$$

46. შექმენით WINDOWS აპლიკაცია, რომელიც TextBox1-ში მოცემულ სტრიქონის შუაში სვამს TextBox2-ში მოცემულ სტრიქონს და შედეგს გამოიტანს TextBox3-ში;
47. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს კენტინდექსებიან ელემენტთა შორის უდიდესს და შედეგს გამოიტანს TextBox-ში;
48. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq 0 \\ a, & (2n-1)a < x \leq 2na \\ 0, & 2na < x \leq (2n+1)a, n = 1, 2, 3, \dots \end{cases}$$

49. შექმენით WINDOWS აპლიკაცია, რომელიც TextBox1-ში მოცემული სტრიქონის პირველი სიტყვას მიუწერს TextBox2-ში მოცემულ სტრიქონს მარჯვნიდან, ხოლო მეორეს - მარცხნიდან და შედეგს გამოიტანს TextBox3-ში;
50. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს ლუწინდექსებიან ელემენტთა შორის უდიდესს და შედეგს გამოიტანს TextBox-ში;
51. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq 0 \\ \sin x, & 0 < x \leq b \\ e^{-(x-\pi/2)}, & b < x \leq d \\ \frac{\sin(b)}{a}(x-d)x, & d < x \leq d+a \\ 0, & x > d+a \end{cases}$$

52. შექმენით WINDOWS აპლიკაცია, რომელიც TextBox1-ში მოცემული სტრიქონის ბოლო სიტყვას მიუწერს TextBox2-ში მოთავსებულ სტრიქონს მარჯვნიდან და შედეგს გამოიტანს TextBox3-ში;
53. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს ნულოვანი ელემენტების რაოდენობას და შედეგს გამოიტანს TextBox-ში;
54. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} b, & x \leq -a \\ \frac{h}{a^2} \times x^2 - h, & -a < x \leq a \\ b, & x > a \end{cases}$$

55. შექმენით WINDOWS აპლიკაცია, რომელიც TextBox1-ში მოცემულ სტრიქონს მიუწერს TextBox2-ში მოთავსებულ სტრიქონს მარჯვნიდან და მარცხნიდან და შედეგს გამოიტანს TextBox3-ში;
56. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს 1-ის ტოლ ელემენტთა რაოდენობას და შედეგს გამოიტანს TextBox-ში;

57. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წვერების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{-h}{a^2} \times x^2 + h, & -a < x \leq a \\ 0, & x > a \end{cases}$$

58. შექმენით WINDOWS აპლიკაცია, რომელიც დაადგინეთ, ემთხვევა თუ არა TextBox1-ში შემაგალი სიტყვები TextBox2-ის სტრიქონის სიტყვებს და შედეგს გამოიტანს TextBox3-ში;

59. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს წვერების ნამრავლს და შედეგს გამოიტანს TextBox-ში;

60. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წვერების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{-h}{a} \times (x+a), & -a < x \leq 0 \\ \frac{h}{a} \times (x-a), & 0 < x \leq a \\ \sin(x-a-\frac{\pi}{2}), & a < x \leq a+\pi \\ 0, & x > a+\pi \end{cases}$$

61. შექმენით WINDOWS აპლიკაცია, რომელიც TextBox1-ში მოცემულ სტრიქონს მიუწერს TextBox2-ის პირველ სიტყვას მარჯვნიდან და შედეგს გამოიტანს TextBox3-ში;

62. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს დადებითი წევრების ნამრავლს და შედეგს გამოიტანს TextBox-ში;

63. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{d-h}{a^2} \times x^2 - d, & -a < x \leq a \\ 0, & x > a \end{cases}$$

64. შექმენით WINDOWS აპლიკაცია, რომელიც TextBox1-ში მოცემულ სტრიქონს მიუწერს TextBox2-ის პირველ სიტყვას მარცხნიდან და შედეგს გამოიტანს TextBox3-ში;

65. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს უარყოფითი წევრების ნამრავლს და შედეგს გამოიტანს TextBox-ში;

66. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} he^{-a}, & x \leq -a \\ he^x, & -a < x \leq 0 \\ he^{-x}, & 0 < x \leq a \\ e^{-a}, & x > a \end{cases}$$

67. შექმენით WINDOWS აპლიკაცია, რომელიც TextBox1-ში მოცემულ სტრიქონს მიუწერს TextBox2-ის ბოლო სიტყვა მარჯვნიდან და შედეგს გამოიტანს TextBox3-ში;

68. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს ლუწინდექსიანი ელემენტების ნამრავლს და შედეგს გამოიტანს TextBox-ში;
69. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} h, x \leq -a \\ -h, -a < x \leq a \\ h, a < x \leq 2a \\ \frac{d}{c}(x-2a), 2a < x \leq 2a+c \\ 0, x > 2a+c \end{cases}$$

70. შექმენით WINDOWS აპლიკაცია, რომელიც TextBox1-ში მოცემულ სტრიქონს მიუწერს TextBox2-ის ბოლო სიტყვას მარცხნიდან და შედეგს გამოიტანს TextBox3-ში;
71. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს მოცემული რიცხვების კვადრატების ჯამს და შედეგს გამოიტანს TextBox-ში;
72. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს , ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} 0, & x \leq -a \\ \frac{-h}{a} \times (x+a), & -a < x \leq 0 \\ \frac{h}{a} (x-a), & 0 < x \leq 2a \\ \operatorname{tg}(x-2a), & 2a < x \leq 3a \\ h, & x > 3a \end{cases}$$

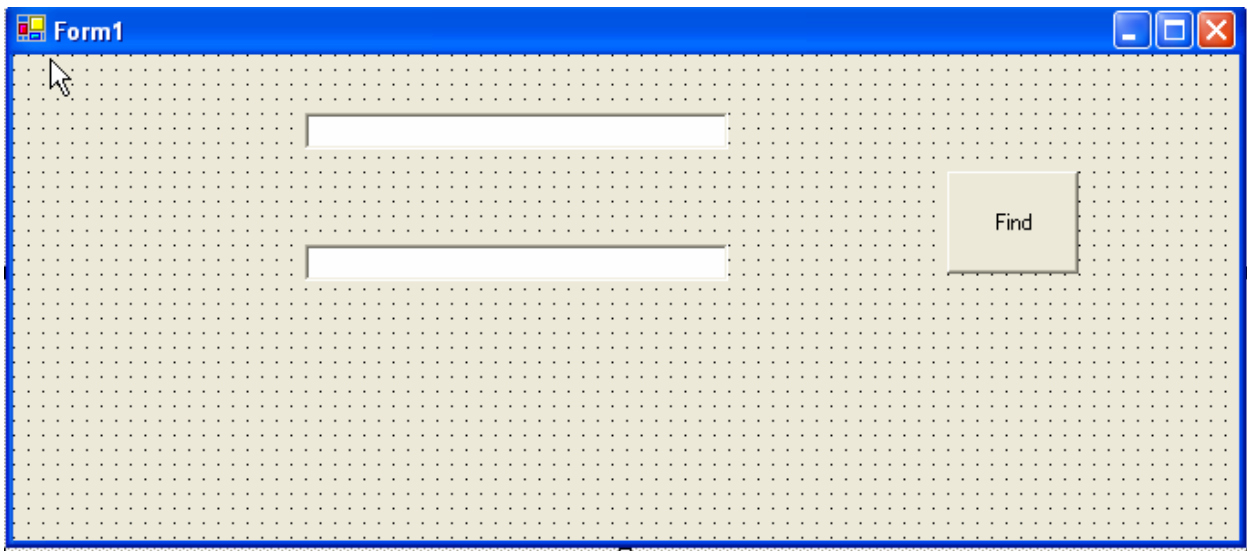
73. შექმენით WINDOWS აპლიკაცია, რომელიც TextBox1-ში და TextBox2-ში მოცემული სტრიქონებიდან დააფორმირებს ახალ სტრიქონს მოცემული სტრიქონების სიტყვების თანმიმდევრობითი მონაცვლებით, ე. ი პირველი სტრიქონის პირველი სიტყვა, შემდეგ მეორე სტრიქონის მეორე სიტყვა და ა.შ. და შედეგს გამოიტანს TextBox3-ში;

74. შექმენით WINDOWS აპლიკაცია, რომელიც ListBox-ში მოცემულ ნამდვილ რიცხვთა მიმდევრობიდან (a_1, a_2, \dots, a_n) განსაზღვრავს $\sqrt{a_1 + \sqrt{a_2 \sqrt{a_3 + \dots + \sqrt{a_n}}}}$ და შედეგს გამოიტანს TextBox-ში;

75. შექმენით WINDOWS აპლიკაცია, რომელიც შემდეგი გამოსახულების წევრების TextBox-ებით შეყვანის საშუალებას მოგვცემს, ხოლო შედეგს Button ღილაკის დაჭერისას სტრიქონის სახით დაამატებს ListBox-ში:

$$Y = \begin{cases} h, & x \leq -R-a \\ \frac{h}{a} \times (x-R), & -R-a < x \leq -R \\ \sqrt{R^2 - x^2}, & -R < x \leq R \\ -\sqrt{R^2 - (x-2R)^2}, & R < x \leq 3R \\ \frac{h}{a} (x-3R), & 3R < x \leq 3R+a \\ h, & x > 3R \end{cases}$$

76. მოცემული ფორმის და პროგრამის ვერიფიკაციის საფუძველზე აღწერეთ თუ რას აკეთებს იგი.



```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;

namespace bileTi_III_nawili_nimuSi_2
{
    /// <summary>
    /// Summary description for Form1.
    /// </summary>
    public class Form1 : System.Windows.Forms.Form
    {
        private System.Windows.Forms.TextBox textBox1;
        private System.Windows.Forms.TextBox textBox2;
        private System.Windows.Forms.Button button1;
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.Container components = null;

        public Form1()
        {
            //
            // Required for Windows Form Designer support
            //
            InitializeComponent();

            //
            // TODO: Add any constructor code after InitializeComponent
            //
        }
    }
}
```

```

/// <summary>
/// Clean up any resources being used.
/// </summary>
protected override void Dispose( bool disposing )
{
    if( disposing )
    {
        if (components != null)
        {
            components.Dispose();
        }
        base.Dispose( disposing );
    }
}

#region Windows Form Designer generated code
/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
private void InitializeComponent()
{
    this.textBox1 = new System.Windows.Forms.TextBox();
    this.textBox2 = new System.Windows.Forms.TextBox();
    this.button1 = new System.Windows.Forms.Button();
    this.SuspendLayout();
    //
    // textBox1
    //
    this.textBox1.Location = new System.Drawing.Point(160, 32);
    this.textBox1.Name = "textBox1";
    this.textBox1.Size = new System.Drawing.Size(232, 20);
    this.textBox1.TabIndex = 0;
    this.textBox1.Text = "";
    //
    // textBox2
    //
    this.textBox2.Location = new System.Drawing.Point(160, 104);
    this.textBox2.Name = "textBox2";
    this.textBox2.Size = new System.Drawing.Size(232, 20);
    this.textBox2.TabIndex = 1;
    this.textBox2.Text = "";
    //
    // button1
    //
    this.button1.Location = new System.Drawing.Point(512, 64);
    this.button1.Name = "button1";
    this.button1.Size = new System.Drawing.Size(72, 56);
    this.button1.TabIndex = 2;
    this.button1.Text = "Find";
    this.button1.Click += new
System.EventHandler(this.button1_Click);
    //
    // Form1
    //
    this.AutoScaleBaseSize = new System.Drawing.Size(5, 13);
    this.ClientSize = new System.Drawing.Size(672, 266);
    this.Controls.Add(this.button1);
    this.Controls.Add(this.textBox2);
    this.Controls.Add(this.textBox1);
    this.Name = "Form1";
    this.Text = "Form1";
    this.ResumeLayout(false);
}

```



```

}
#endregion

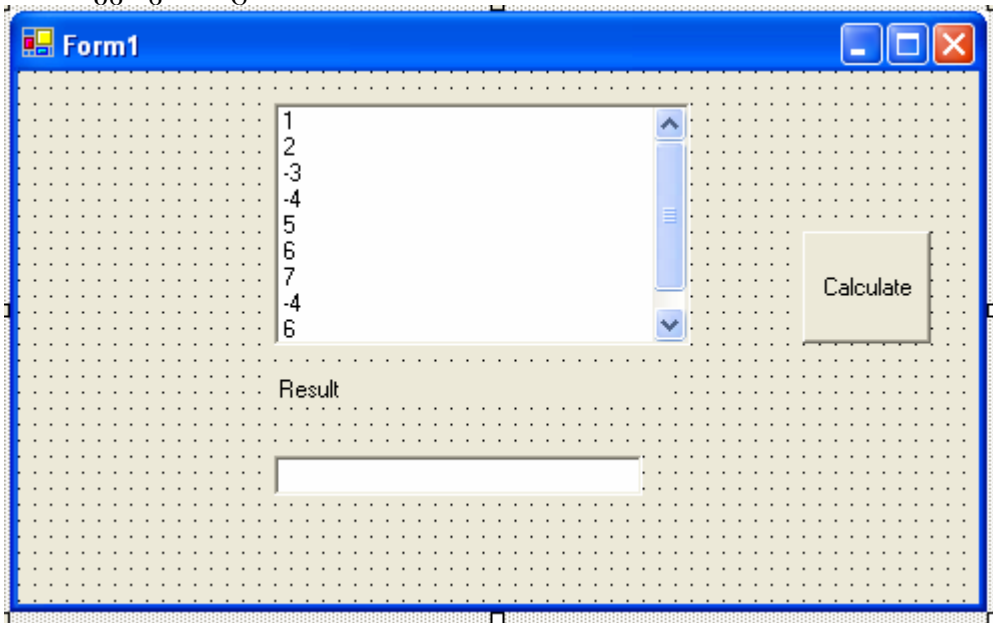
/// <summary>
/// The main entry point for the application.
/// </summary>
[STAThread]
static void Main()
{
    Application.Run(new Form1());
}

private void button1_Click(object sender, System.EventArgs e)
{
    string s1; int k=0;s1=this.textBox1.Text;

char[] ganmacalkavebeli=new char[] { ' ' }; string striqoni=s1;
foreach (string qvestriqoni in striqoni.Split(ganmacalkavebeli))
{
    if(qvestriqoni=="Keti")k++;
}
        this.textBox2.Text=k.ToString();
}
}
}

```

77. მოცემული ფორმის და პროგრამის ვერიფიკაციის საფუძველზე აღწერეთ თუ რას აკეთებს იგი.



```

using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;

namespace bileTi_III_nawili_nimuSi_3
{
    /// <summary>
    /// Summary description for Form1.
    /// </summary>
    public class Form1 : System.Windows.Forms.Form

```

```

{
    private ListBox listBox1;
    private Label label1;
    private TextBox textBox1;
    private Button button1;
    /// <summary>
    /// Required designer variable.
    /// </summary>
    private System.ComponentModel.Container components = null;

    public Form1()
    {
        //
        // Required for Windows Form Designer support
        //
        InitializeComponent();

        //
        // TODO: Add any constructor code after InitializeComponent
        //
    }

    /// <summary>
    /// Clean up any resources being used.
    /// </summary>
    protected override void Dispose( bool disposing )
    {
        if( disposing )
        {
            if (components != null)
            {
                components.Dispose();
            }
        }
        base.Dispose( disposing );
    }

    #region Windows Form Designer generated code
    /// <summary>
    /// Required method for Designer support - do not modify
    /// the contents of this method with the code editor.
    /// </summary>
    private void InitializeComponent()
    {
        this.listBox1 = new System.Windows.Forms.ListBox();
        this.label1 = new System.Windows.Forms.Label();
        this.textBox1 = new System.Windows.Forms.TextBox();
        this.button1 = new System.Windows.Forms.Button();
        this.SuspendLayout();
        //
        // listBox1
        //
        this.listBox1.Items.AddRange(new object[] {
"1", "2", "-3", "-4", "5", "6", "7", "-4", "6", "9"});
        this.listBox1.Location = new System.Drawing.Point(128, 16);
        this.listBox1.Name = "listBox1";
        this.listBox1.Size = new System.Drawing.Size(208, 121);
        this.listBox1.TabIndex = 0;
        //
        // label1
        //
        this.label1.Location = new System.Drawing.Point(128, 152);

```

```

        this.labell1.Name = "labell1";
        this.labell1.Size = new System.Drawing.Size(200, 16);
        this.labell1.TabIndex = 1;
        this.labell1.Text = "Result";
        //
        // textBox1
        //
        this.textBox1.Location = new System.Drawing.Point(128, 192);
        this.textBox1.Name = "textBox1";
        this.textBox1.Size = new System.Drawing.Size(184, 20);
        this.textBox1.TabIndex = 2;
        this.textBox1.Text = "";
        //
        // button1
        //
        this.button1.Location = new System.Drawing.Point(392, 80);
        this.button1.Name = "button1";
        this.button1.Size = new System.Drawing.Size(64, 56);
        this.button1.TabIndex = 3;
        this.button1.Text = "Calculate";
        this.button1.Click += new
System.EventHandler(this.button1_Click);
        //
        // Form1
        //
        this.AutoScaleBaseSize = new System.Drawing.Size(5, 13);
        this.ClientSize = new System.Drawing.Size(480, 266);
        this.Controls.Add(this.button1);
        this.Controls.Add(this.textBox1);
        this.Controls.Add(this.labell1);
        this.Controls.Add(this.listBox1);
        this.Name = "Form1";
        this.Text = "Form1";
        this.ResumeLayout(false);

    }
#endregion

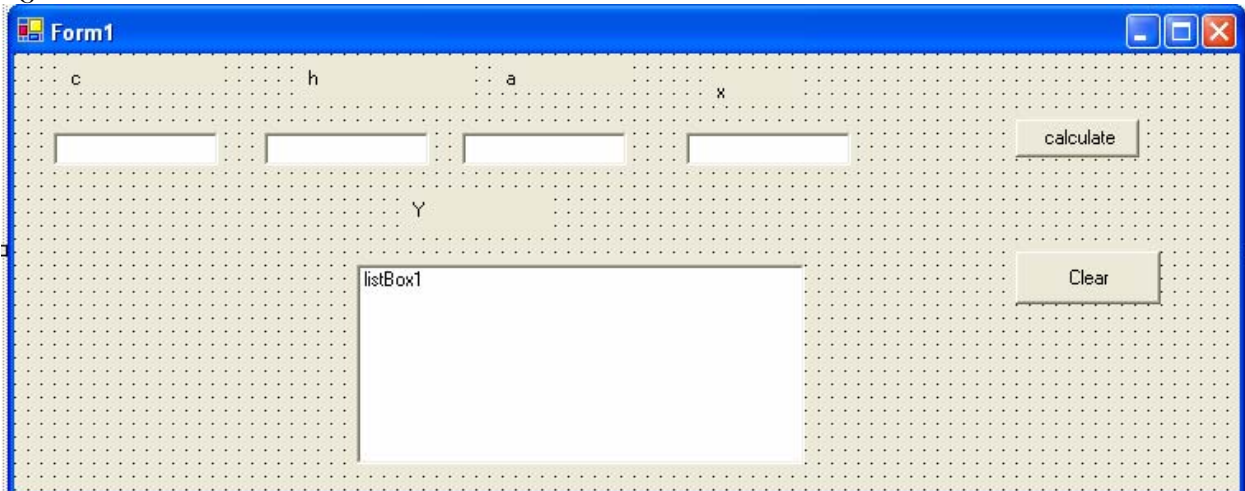
/// <summary>
/// The main entry point for the application.
/// </summary>
[STAThread]
static void Main()
{
    Application.Run(new Form1());
}

private void button1_Click(object sender, System.EventArgs e)
{
    double S=0;
    foreach(string x in listBox1.Items)
    {
        if(double.Parse(x)<0)
            S+=double.Parse(x);
    }

    this.textBox1.Text=S.ToString();
}
}
}

```

78. მოცემული ფორმის და პროგრამის ვერიფიკაციის საფუძველზე აღწერეთ თუ რას აკეთებს იგი.



```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;

namespace bileTi_III_nawili_nimuSi
{
    /// <summary>
    /// Summary description for Form1.
    /// </summary>

    public class Form1 : System.Windows.Forms.Form
    {
        private System.Windows.Forms.Button button1;
        private System.Windows.Forms.TextBox textBox1;
        private System.Windows.Forms.TextBox textBox2;
        private System.Windows.Forms.TextBox textBox3;
        private System.Windows.Forms.TextBox textBox4;
        private System.Windows.Forms.Label label1;
        private System.Windows.Forms.Label label2;
        private System.Windows.Forms.Label label3;
        private System.Windows.Forms.Label label4;
        private System.Windows.Forms.Label label5;
        private System.Windows.Forms.ListBox listBox1;
        private System.Windows.Forms.Button button2;
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.Container components = null;

        public Form1()
        {
            //
            // Required for Windows Form Designer support
            //
            InitializeComponent();

            //
            // TODO: Add any constructor code after InitializeComponent
            //
        }
    }
}

```

```

}

/// <summary>
/// Clean up any resources being used.
/// </summary>
protected override void Dispose( bool disposing )
{
    if( disposing )
    {
        if (components != null)
        {
            components.Dispose();
        }
        base.Dispose( disposing );
    }
}

#region Windows Form Designer generated code
/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
private void InitializeComponent()
{
    this.button1 = new System.Windows.Forms.Button();
    this.textBox1 = new System.Windows.Forms.TextBox();
    this.textBox2 = new System.Windows.Forms.TextBox();
    this.textBox3 = new System.Windows.Forms.TextBox();
    this.textBox4 = new System.Windows.Forms.TextBox();
    this.label1 = new System.Windows.Forms.Label();
    this.label2 = new System.Windows.Forms.Label();
    this.label3 = new System.Windows.Forms.Label();
    this.label4 = new System.Windows.Forms.Label();
    this.label5 = new System.Windows.Forms.Label();
    this.listBox1 = new System.Windows.Forms.ListBox();
    this.button2 = new System.Windows.Forms.Button();
    this.SuspendLayout();
    //
    // button1
    //
    this.button1.Location = new System.Drawing.Point(608, 40);
    this.button1.Name = "button1";
    this.button1.TabIndex = 0;
    this.button1.Text = "calculate";
    this.button1.Click += new
System.EventHandler(this.button1_Click);
    //
    // textBox1
    //
    this.textBox1.Location = new System.Drawing.Point(24, 48);
    this.textBox1.Name = "textBox1";
    this.textBox1.TabIndex = 1;
    this.textBox1.Text = "";
    //
    // textBox2
    //
    this.textBox2.Location = new System.Drawing.Point(152, 48);
    this.textBox2.Name = "textBox2";
    this.textBox2.TabIndex = 2;
    this.textBox2.Text = "";
    //
    // textBox3
    //
    this.textBox3.Location = new System.Drawing.Point(272, 48);

```

```

this.textBox3.Name = "textBox3";
this.textBox3.TabIndex = 3;
this.textBox3.Text = "";
//
// textBox4
//
this.textBox4.Location = new System.Drawing.Point(408, 48);
this.textBox4.Name = "textBox4";
this.textBox4.TabIndex = 4;
this.textBox4.Text = "";
//
// label1
//
this.label1.Location = new System.Drawing.Point(32, 8);
this.label1.Name = "label1";
this.label1.Size = new System.Drawing.Size(96, 16);
this.label1.TabIndex = 6;
this.label1.Text = "c";
//
// label2
//
this.label2.Location = new System.Drawing.Point(176, 8);
this.label2.Name = "label2";
this.label2.TabIndex = 7;
this.label2.Text = "h";
//
// label3
//
this.label3.Location = new System.Drawing.Point(296, 8);
this.label3.Name = "label3";
this.label3.Size = new System.Drawing.Size(80, 16);
this.label3.TabIndex = 8;
this.label3.Text = "a";
//
// label4
//
this.label4.Location = new System.Drawing.Point(424, 16);
this.label4.Name = "label4";
this.label4.Size = new System.Drawing.Size(56, 16);
this.label4.TabIndex = 9;
this.label4.Text = "x";
//
// label5
//
this.label5.Location = new System.Drawing.Point(240, 88);
this.label5.Name = "label5";
this.label5.Size = new System.Drawing.Size(88, 24);
this.label5.TabIndex = 10;
this.label5.Text = "Y";
//
// listBox1
//
this.listBox1.Location = new System.Drawing.Point(208, 128);
this.listBox1.Name = "listBox1";
this.listBox1.Size = new System.Drawing.Size(272, 121);
this.listBox1.TabIndex = 11;
//
// button2
//
this.button2.Location = new System.Drawing.Point(608, 120);
this.button2.Name = "button2";
this.button2.Size = new System.Drawing.Size(88, 32);
this.button2.TabIndex = 12;
this.button2.Text = "Clear";

```

```

        this.button2.Click += new
System.EventHandler(this.button2_Click);
        //
        // Form1
        //
        this.AutoScaleBaseSize = new System.Drawing.Size(5, 13);
        this.ClientSize = new System.Drawing.Size(744, 266);
        this.Controls.Add(this.button2);
        this.Controls.Add(this.listBox1);
        this.Controls.Add(this.label5);
        this.Controls.Add(this.label4);
        this.Controls.Add(this.label3);
        this.Controls.Add(this.label2);
        this.Controls.Add(this.label1);
        this.Controls.Add(this.textBox4);
        this.Controls.Add(this.textBox3);
        this.Controls.Add(this.textBox2);
        this.Controls.Add(this.textBox1);
        this.Controls.Add(this.button1);
        this.Name = "Form1";
        this.Text = "Form1";
        this.ResumeLayout(false);

    }
    #endregion

    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main()
    {
        Application.Run(new Form1());
    }

    private void button1_Click(object sender, System.EventArgs e)
    {

        this.listBox1.Items.Add(F(double.Parse(this.textBox1.Text),double.Parse(th
is.textBox2.Text),double.Parse(this.textBox3.Text),double.Parse(this.textBox4.Te
xt)).ToString());

    }
    private static double F(double a,double c,double h,double x)
    {
        if(x<=0)return 0;
        else if (x<Math.PI)return((c+h)/(a*a)*Math.Pow(x-a,2)-h);
        else return 0;
    }

    private void button2_Click(object sender, System.EventArgs e)
    {
        this.listBox1.Items.Clear();
    }
}

```

ლიტერატურა

1. თ. ბახტაძე. ობიექტზე ორიენტირებული პროგრამირების ენა C#. სტუ, თბილისი, 2006.-129 გვ. ISBN 99940-57-20-0.
2. Разработка Windows-приложений на Microsoft visual Basic и Microsoft Visual C# .NET. Учебный курс MCAD MCSD /Пер с англ. –М.: Издательско-торговый дом "Русская Редакция", 2003.-512 стр.:ил. ISBN 5-7502-0222-4.
3. <http://www.microsoft.com/learning/syllabi/en-us/2609Afinal.msp>
4. <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cscon/html/vcoriCStartPage.asp>
5. <http://zone.ni.com/devzone/conceptd.nsf/webmain/80AC25B00E13CEE386256D1A00648A54#1>
6. <http://www.softsteel.co.uk/tutorials/cSharp/contents.html>
7. <http://www.devarticles.com/c/a/C-Sharp/Visual-C-Sharp.NET-Part-1-Introduction-to-Programming-Languages/>
8. http://www.e-trainonline.com/html/visual_c_net.html

იბეჭდება ავტორის მიერ წარმოდგენილი სახით

გადაეცა წარმოებას 29.05.2008. ხელმოწერილია დასაბეჭდად 23.06.2008. ქაღალდის ზომა 60X84 1/8. პირობითი ნაბეჭდი თაბახი 5. ტირაჟი 100 ეგზ.

საგამომცემლო სახლი „ტექნიკური უნივერსიტეტი“, თბილისი, კოსტავას 77

