

**ელექტრონული სწავლების სისტემა დისციპლინაში
„ავტომატიზებული მართვის მოდელები“**

გიორგი გოგიჩაიშვილი, ეკატერინე თურქია, ლილი ჰეტრიაშვილი
საქართველოს ტექნიკური უნივერსიტეტი

რეზიუმე

შემოთავაზებულია სიმრავლეთა თეორიისა და მათემატიკური ლოგიკის დისციპლინის ლაბორატორიული მეცადინეობისათვის შექმნილი პროგრამული პაკეტი. მოცემული კურსი ალგორითმიზაციისა და დაპროგრამების თეორიის შესწავლის საბაზისო ნაწილია, ამდენად ამ დისციპლინის სიღრმისეულ შესწავლას, რასაც უზრუნველყოფს ჩვენს მიერ დამუშავებული სისტემა, დიდი მნიშვნელობა აქვს სტუდენტებისთვის უნარ-ჩვევების ჩამოყალიბებაში, ისეთ საკითხებში, როგორცაა მასივებზე ოპერაციების ჩატარება, რელაციური ალგებრისა და მათემატიკური ლოგიკის ძირითად ოპერაციებზე ზოგადი წარმოდგენის შექმნა და სხვა. სისტემა გამოკვლეულია და რეალიზებულია თანამედროვე პროგრამული ტექნოლოგიებით - .Net C#, UML2-Enterprise Architect. პროგრამული პროდუქტი ტექნიკურად შექმნილია გაფართოებადი არქიტექტურით, რაც სხვა პროგრამულ პაკეტებთან ინტეგრაციის, ახალი ამოცანების დამატებისა და იმპლემენტირების საშუალებას იძლევა.

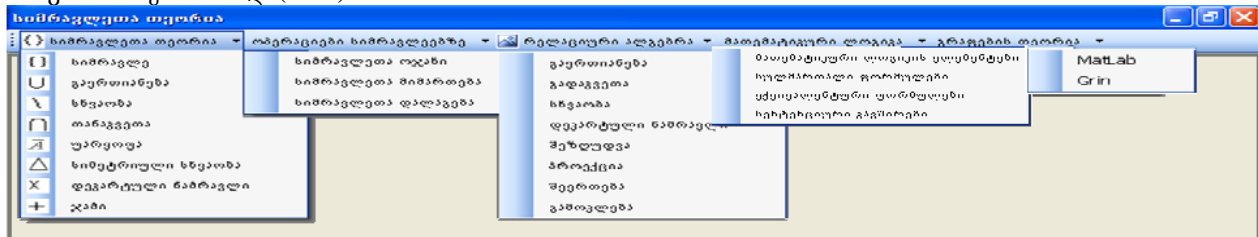
საკანძო სიტყვები: სიმრავლეთა თეორია. რელაციური ალგებრა. მათემატიკური ლოგიკა. ოპერაციები.

1. შესავალი

თანამედროვე საინფორმაციო ტექნოლოგიების განვითარებამ შესაძლებელი გახადა განათლების პრობლემების გადაწყვეტის კონცეფციების, სოციალურ-ფსიქოლოგიური და ფსიქო-ფიზიოლოგიური გამოკვლევების საფუძველზე საზოგადოებისათვის შეთავაზებინა ახალი პროგრამული და ინსტრუმენტული საშუალებანი, რაც „სწავლებას“ მეცნიერების განვითარების სხვადასხვა საფეხურზე გახდიდა ხელმისაწვდომს. სასწავლო პროცესების ინფორმატიზაცია მიზნად ისახავს მომავალ სპეციალისტთა მაღალ დონეზე მომზადებისათვის სწავლებაში დაინერგოს და აქტიურად იქნეს გამოყენებული თანამედროვე საინფორმაციო ტექნოლოგიები. სწავლების ინფორმატიზაციისათვის აუცილებელია თანამედროვე კომპიუტერული ბაზა და საინფორმაციო ტექნოლოგიების უწყვეტად გამოყენება. აკადემიური სექტორის განვითარება ერთ-ერთი უმნიშვნელოვანესი და სტრატეგიული საკითხია საზოგადოებრივი ცხოვრების განვითარებაში. ამ თვალსაზრისით, დღეს დიდი ყურადღება ეთმობა ელექტრონულ/დისტანციურ სწავლებასა და სასწავლო სექტორში ინფორმატიზაციის პროცესის დანერგვას. მართვის ავტომატიზებული სისტემების ფუნქციონალური ქვესისტემების ინფორმაციული, ალგორითმული და მათემატიკური უზრუნველყოფის დაპროექტებისას მნიშვნელოვან როლს ასრულებს ლოგიკური და გრაფული მოდელები [1].

2. ძირითადი ნაწილი

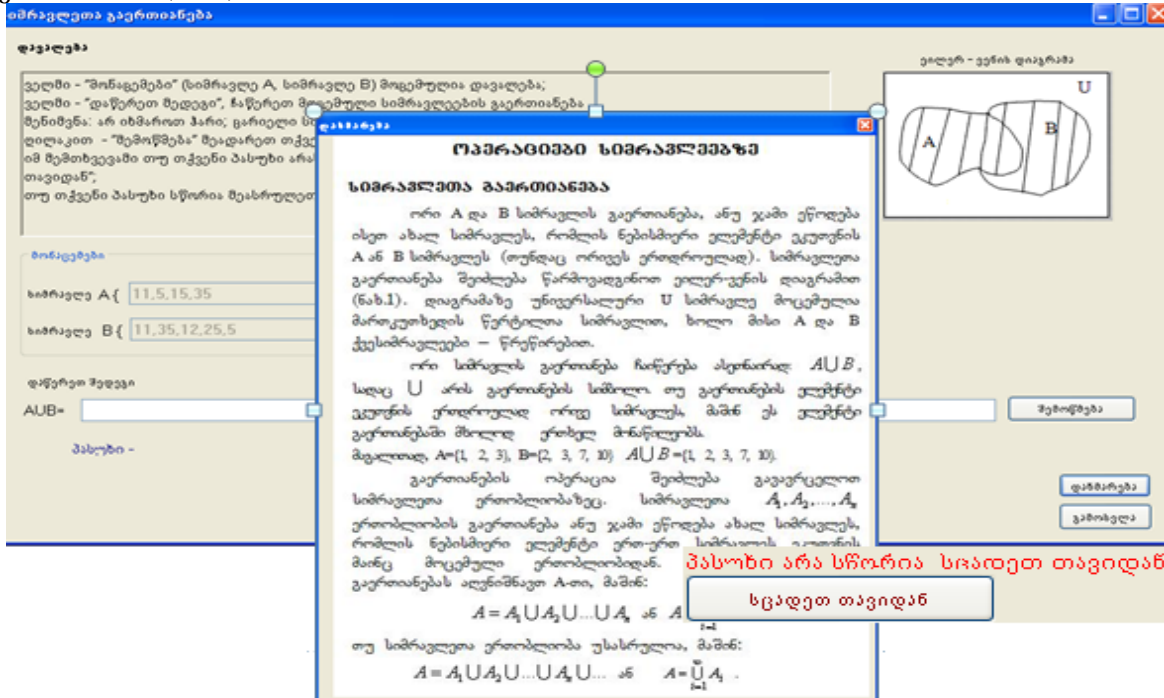
რეალიზებული სისტემა მომხმარებელს საშუალებას აძლევს ეტაპობრივად სირთულის მიხედვით გადაჭრას დასმული ამოცანები, პრობლემის წარმოქმნის შემთხვევაში მიიღოს სისტემის მიერ შეთავაზებული კომპეტენტური რეკომენდაციები და საბოლოოდ მიიღოს ოპტიმალური შედეგი. პროგრამული პაკეტი მოიცავს სიმრავლეთა თეორიის ძირითადი საკითხების – სიმრავლეთა ოპერაციების, რელაციური ალგებრისა და მათემატიკური ლოგიკის პრაქტიკული შესწავლის შესაძლებლობებს. ყოველი ძირითადი საკითხი დაყოფილია შემადგენელი ოპერაციების ცალკეულ დიალოგებად, რომელთა მუშაობის პრინციპი აგებულია სასწავლო პროგრამის შესაბამისად (ნახ.1).



ნახ.1. სისტემის მთავარი მენიუს ფრაგმენტი

სისტემაში წარმოდგენილი ამოცანები და მათი გადაწყვეტის ფუნქციები რთულდება ეტაპობრივად, რაც სტუდენტებს საგნის სიღრმისეულად შესწავლის საშუალებას აძლევს. როგორც ცნობილია, სიმრავლე მოცემულია რაიმე კანონის თანახმად წარმოდგენილი ელემენტების სახით, რომლის დახმარებითაც შესაძლებელია ამ ელემენტთა განსაზღვრა.

სისტემის მთავარი მენიუს პირველ ჩანართში „სიმრავლეთა თეორია“ წარმოდგენილია სიმრავლისა და სიმრავლის შემადგენელ ელემენტთა განსაზღვრის ძირითადი ფუნქციები, რომლის ჯგუფშიც შედის სიმრავლის განსაზღვრის, სიმრავლეთა გაერთიანების, სხვაობის, თანაკვეთის, უარყოფის, სიმეტრიული სხვაობის, დეკარტული ნამრავლისა და ჯამის ოპერაციების ამოცანები. თითოეული დიალოგური ფორმა დამატებით შეიცავს დავალების შესრულების აღწერის, დახმარებისა და ოპერაციის შესაბამისი მოდელის ინფორმაციას ეილერ-ვენის დიაგრამის სახით (ნახ.2).



ნახ.2.

```

Logika - Microsoft Visual Studio
Form_jami.cs [Design]* Start Page Form_logika_start.cs Form_logika_kvashirebi.cs
ni
private void button_tavidan_Click(object sender, EventArgs e)
{
    textBox_pasuxiA.Text = "";
    textBox_sedegiA.Text = "";
    textBox_A.Text = "";
    textBox_B.Text = "";
    label_pasuxi.Text = "პასუხი -";
    this.label_pasuxi.ForeColor = System.Drawing.Color.Navy;

    groupBox_shedegi.Visible = false;
    if (next == 0 && pas == 0)
    {
        System.Random obieqti = new System.Random();
        String result = "";
        for (int i = 0; i < 2; i++)
        {
            int rn = obieqti.Next(1, 90);
            result += rn.ToString() + ",";
        }
        result = result.Remove(result.Length - 1, 1);
        String A = simravle(result);
        textBox_A.Text = A;

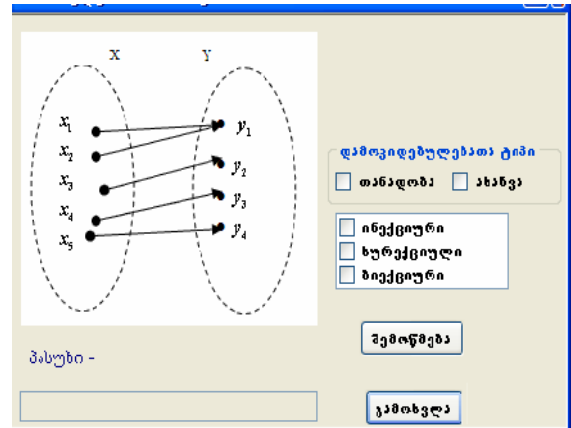
        System.Random obieqtib = new System.Random();
        String resultb = "";
    }
}
    
```

ნახ.3

პრაქტიკული თვალსაზრისით სისტემა ატარებს როგორც შემცენების ხასიათს, ისე ტესტირების ფორმებს: თითოეული საკითხის შესწავლა მოწმდება ეტაპობრივად გართულებული ციკლით; სისტემა ავტომატიზებულ რეჟიმში ამოწმებს შესრულებული დავალების სისწორეს. იმ შემთხვევაში თუ მიმდინარე ეტაპის პასუხი სწორია, სისტემა გადადის შემდეგ გართულებულ დავალებაზე, წინააღმდეგ შემთხვევაში სისტემას გამოაქვს სწორი პასუხი და ავტომატურად უგენერირებს ახალ დავალებას იმ ეტაპზე, რომელზეც ვერ მოხდა სწორი პასუხის მიღება.

ტექნიკური თვალსაზრისით ამოცანათა მრავალსახეობისთვის გამოყენებულია შემთხვევითი ფორმირების RUNDOM ფუნქცია (ნახ.3), რომელიც მომხმარებელს ოპერაციების წარმოების საშუალებას აძლევს, სხვადასხვა ტიპის (სიმბოლოები, ციფრები, სიტყვები) ელემენტებისაგან შედგენილ სიმრავლეზე [1, 2].

მთავარი მენიუს ჩანართში „სიმრავლეთა თეორია“ ყოველი ოპერაციის შესრულებას მომხმარებელს უიოლებს როგორც სისტემის მიერ შეთავაზებული გრაფიკული გამოსახულება, ასევე მოკლე თეორიული მითითებები. ამავე დროს სტუდენტი ეჩვევა როგორც სისტემასთან მუშაობას, ისე იმ მნიშვნელოვანი უნარჩვევების გამომუშავებას, რომელიც აუცილებელი და საჭირო პირობაა მომდევნო დავალებზე გადასასვლელად. მომდევნო დავალებაა „ოპერაციები სიმრავლეებზე“. ამ დავალების შესრულებისას სტუდენტი ასრულებს ისეთ ფუნქციებს, როგორცაა: სიმრავლეთა ოჯახი, სიმრავლეთა დალაგება, სიმრავლეთა მიმართება. ამ ფუნქციათა გამოყენებით სტუდენტი ასრულებს ამოცანებს, სადაც წარმოდგენილია ორ ობიექტს შორის ბინარული დამოკიდებულება (ნახ.4). შემდეგი ჩანართი განკუთვნილია რელაციური ალგებრის ძირითადი პრინციპების შესასწავლად (პროექცია, გადაკვეთა, გამოკლება, შეერთება, შეზღუდვა, დეკარტული ნამრავლი, გაყოფა). პროგრამის მნიშვნელოვან ღირებულებას წარმოადგენს ისეთი მეთოდების გამოყენება, როგორცაა: ცხრილური სტრუქტურის ფუნქციები DATATABLE LINQ expression, და Query AsEnumerable() Tables (ნახ.5), რაც SQL-ის DML ენის გამოყენების საშუალებას იძლევა ობიექტ-ორიენტირებული API გარემოში [2,3].



ნახ.4

```
int scori_pasuxi_Table(int state)
{
    groupBox_pasuxi.Visible = true;
    DataSet ds = new DataSet();
    ds.Locale = CultureInfo.InvariantCulture;

    DataTable tableA = korect.return_dat
    DataTable tableB = korect.return_dat

    DataTable table_scori = new DataTabl
    table_scori = tableA.Copy();
    table_scori.Merge(tableB);
    table_scori.Rows.Clear();

    if (state == 0)
    {
        var query =
            tableA.AsEnumerable().Join(t
            order => order.Field<String>
            contact => contact.Field<STR
            (order, contact) => new
            {
                TableOID = order.Field<S
                Table1ID = order.Field<S
                Table2ID = order.Field<S
                Table3ID = contact.Field<STR
            };
    }
}
```

ნახ.5

სისტემაში მნიშვნელოვანი ყურადღება ეთმობა აგრეთვე მათემატიკური ლოგიკის ამოცანებს და მათთან დაკავშირებულ ლოგიკის ელემენტებს (ნახ.6) (ლოგიკური გამოყვანა, სენტენციური კავშირები, სულმართალი ფორმულები და ა.შ.).

3. დასკვნა

სისტემის არქიტექტურა იძლევა სხვა პროგრამულ პაკეტებთან ინტეგრაციისა და ახალი ამოცანების დამატების გაფართოების შესაძლებლობებს (დისციპლინის „ავტომატიზებული მართვის მოდელები“ ცვლილებისა და განვითარების შესაბამისად). სისტემის მოდელი აღწერილია და გამოკვლეულია UML2 (Enterprise Architect პროდუქტზე) ტექნოლოგიით, ხოლო პროგრამული პაკეტი რეალიზებულია .NET C# ენაზე. სისტემის გაფართოების თვალსაზრისით ახალ ვერსიებში იმპლემენტირებული იქნება და ყურადღება მიექცევა შემდეგი საკითხების დახვეწას: საშუალო ადგილებად (სტუდენტი-ლექტორი) ფუნქციონირების შესაძლებლობა; დავალებათა შერჩევა სირთულის ვარიანტების მიხედვით; სტუდენტის ავტომატიზებული შეფასების ფუნქციონალის ჩართვა.

