



TASODIFIY SONLARGA OIDALGORITMLAR ULARNING MATEMATIK ASOSI VA DASTURLASHDAGI QO'LLANILISHI

Farmonov Sherzodbek Raxmonjonovich

Farg'ona davlat universiteti amaliy matematika va
informatika kafedrası katta o'qituvchisi
farmonovsh@gmail.com

Isaqova Sanobarxon Shuhratjon qizi

Farg'ona davlat universiteti talabasi
isaqovasanoobar67@gmail.com
<https://doi.org/10.5281/zenodo.14498630>

Annotatsiya. Ushbu maqolada tasodifiy sonlar yaratish algoritmlari, ularning matematik asosi va dasturlashdagi qo'llanilishi o'rganilgan. Ayniqsa, C# dasturlash tilida tasodifiy sonlar yaratish va ularning amaliy masalalarda qo'llanilishi haqida batafsil ma'lumot berilgan. Masalalar misolida algoritmning ishlash jarayoni va kod namunasi keltiriladi.

Kalit so'zlar: Tasodifiy sonlar, algoritmlar, C# dasturlash tili, dasturlash, matematik asoslar.

Аннотация. В данной статье рассматриваются алгоритмы генерации случайных чисел, их математические основы и применение в программировании. Особое внимание уделяется генерации случайных чисел на языке программирования C# и их применению в практических задачах. В статье приводятся примеры работы алгоритмов и код.

Ключевые слова: Случайные числа, алгоритмы, язык программирования C#, программирование, математические основы.

Annotation. This article explores algorithms for generating random numbers, their mathematical basis, and their applications in programming. Special attention is given to generating random numbers in the C# programming language and their use in practical problems. Examples of the algorithms and code implementations are provided.

Keywords: Random numbers, algorithms, C# programming language, programming, mathematical basis.

Tasodifiy sonlar dasturlashda ko'plab sohalarda, jumladan, simulyatsiya, statistika, o'yinlar, shifrlash, va testlash kabi jarayonlarda keng qo'llaniladi. Tasodifiy sonlar algoritmlarini ishlab chiqish dasturchilarga ko'plab imkoniyatlar yaratadi. C# dasturlash tilida tasodifiy sonlar yaratish va ulardan samarali foydalanish uchun turli xil metodlar mavjud. Ushbu maqolada C# tilida tasodifiy sonlarni yaratish va ularni ishlatish uchun asosiy algoritmlar va metodlar haqida so'z yuritamiz.



Tasodifiy Sonlar Generatsiyasi

C# dasturlash tilida tasodifiy sonlar yaratish uchun System.Random sinfidan foydalaniladi. Bu sinf tasodifiy butun sonlar, o'nli sonlar (float, double) va boshqa turdagi tasodifiy ma'lumotlarni yaratish imkonini beradi. Random sinfi vaqt va boshqa parametrlar asosida tasodifiy sonlar yaratish uchun ishlatiladi.

1. Random Sinfini Ishlatish

C# dasturida tasodifiy son yaratish uchun eng oddiy metod quyidagicha ko'rinadi:

```
C#
using System;

class Program
{
    static void Main()
    {
        Random random = new Random(); // Random obykti yaratish
        int randomInt = random.Next(1, 101); // 1 va 100 orasidagi tasodifiy
butun son
        double randomDouble = random.NextDouble(); // 0 va 1 orasidagi
tasodifiy real son
        Console.WriteLine("Tasodifiy butun son: " + randomInt);
        Console.WriteLine("Tasodifiy real son: " + randomDouble);
    }
}
```

Yuqoridagi kodda:

- random.Next(1, 101) — bu metod 1 va 100 orasidagi tasodifiy butun sonni generatsiya qiladi.
- random.NextDouble() — bu metod 0 va 1 orasidagi tasodifiy real sonni generatsiya qiladi.

2. Random Sinfini Avtorizatsiyalash

Random sinfi, odatda, bir xil boshlang'ich nuqtaga ega bo'lgan (seed) tasodifiy sonlar yaratadi. Bu esa, agar siz bir xil boshlang'ich nuqtani takrorlasangiz, har safar bir xil natija olishingizga olib kelishi mumkin. Agar tasodifiy sonlarning chiqishini yanada tasodifiy qilishni istasangiz, siz tasodifiy boshlang'ich nuqta (seed) belgilashingiz mumkin.

C#

```
Random randomWithSeed = new Random(DateTime.Now.Millisecond); //  
Vaqtga asoslangan seed
```

```
int randomIntWithSeed = randomWithSeed.Next(1, 101);
```

```
Console.WriteLine("Tasodifiy son (seed bilan): " + randomIntWithSeed);
```

Bu usulda `DateTime.Now.Millisecond` vaqtning millisekundiga asoslanib tasodifiy boshlang'ich nuqta (seed) aniqlanadi va har safar dastur ishga tushganda yangi tasodifiy sonlar olinadi.

Tasodifiy Sonlarning Turli Turlari

C# da tasodifiy sonlar faqat butun sonlar bilan cheklanmaydi. Quyida turli xil tasodifiy sonlar yaratish metodlari keltirilgan.

1. Butun Sonlar (Integer)

Next metodi yordamida tasodifiy butun sonlar yaratish mumkin:

```
C#
```

```
Random random = new Random();
```

```
int randomInt = random.Next(50, 101); // 50 va 100 orasidagi tasodifiy  
son
```

2. Real Sonlar (Floating Point Numbers)

`NextDouble` metodi 0 va 1 orasidagi tasodifiy haqiqiy sonlarni yaratadi. Agar kerakli diapazonda tasodifiy son yaratmoqchi bo'lsangiz, uni o'zgartirishingiz mumkin:

```
C#
```

```
double randomReal = random.NextDouble() * 100; // 0 va 100 orasidagi  
tasodifiy haqiqiy son
```

3. Boolean (Mantiqiy Sonlar)

Mantiqiy qiymatlarni tasodifiy tarzda yaratish uchun `Next(0, 2)` metodidan foydalanish mumkin, bu 0 yoki 1 ni qaytaradi, so'ngra mantiqiy qiymatga o'zgartiriladi:

```
C#
```

```
bool randomBool = random.Next(0, 2) == 0; // Tasodifiy true yoki false
```

```
Console.WriteLine("Tasodifiy mantiqiy qiymat: " + randomBool);
```

4. Elementlarni Tasodifiy Tanlash

Ba'zan siz tasodifiy elementlarni massivdan yoki ro'yxatdan tanlashingiz kerak bo'ladi. Buni amalga oshirish uchun `Next` metodini ishlatib, massiv yoki ro'yxatdagi indeksni tasodifiy tanlashingiz mumkin:

```
C#
```

```
string[] colors = { "Red", "Green", "Blue", "Yellow", "Pink" };
```



```
string randomColor = colors[random.Next(colors.Length)]; // Tasodifiy  
rang tanlash
```

```
Console.WriteLine("Tasodifiy rang: " + randomColor);
```

Tasodifiy Sonlarni Boshqa Algoritmarda Qo'llash

Tasodifiy sonlar o'zgaruvchilarni yaratishda, testlashda yoki murakkab algoritmlar (masalan, genetik algoritmlar, shifrlash tizimlari) yaratishda qo'llaniladi. Quyida ba'zi misollar keltirilgan.

1. Monte Carlo Simulyatsiyasi

Monte Carlo metodlari simulyatsiya qilish uchun tasodifiy sonlardan foydalanadi. Masalan, pi sonini hisoblash uchun Monte Carlo metodini qo'llash:

```
C#
```

```
using System;
```

```
class Program
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        Random random = new Random();
```

```
        int insideCircle = 0;
```

```
        int totalPoints = 1000000;
```

```
        for (int i = 0; i < totalPoints; i++)
```

```
        {
```

```
            double x = random.NextDouble();
```

```
            double y = random.NextDouble();
```

```
            if (x * x + y * y <= 1) // Uchun x va y koordinatalari doiraga kirishi
```

```
kerak
```

```
            {
```

```
                insideCircle++;
```

```
            }
```

```
        }
```

```
        double piEstimate = 4.0 * insideCircle / totalPoints;
```

```
        Console.WriteLine("Monte Carlo metodi bilan pi: " + piEstimate);
```

```
    }
```

```
}
```

Bu kod pi ni taxminan hisoblash uchun tasodifiy nuqtalar generatsiyasidan foydalanadi.





2. Tasodifiy Yechimlar Topish (Simulated Annealing)

Simulyatsiya qilingan annealing (Simulated Annealing) algoritmi optimizatsiya masalalarini yechishda tasodifiy sonlar va ularning harakatini ishlatadi.

3. Genetik Algoritmalar

Genetik algoritmalar ham tasodifiy elementlarni o'z ichiga oladi, masalan, mutatsiya, krossover (crossover) va selektsiya jarayonlari.

C# dasturlash tilida tasodifiy sonlarga oid masala va yechim

Masala: 1 dan 100 gacha bo'lgan tasodifiy sonlar orasidan 10 ta son tanlab, ularning o'rtacha qiymatini hisoblash.

Kod:

```
Using System;
class Program
{
    static void Main(string[] args)
    {
        Random random = new Random();          int[] randomNumbers = new
int[10];
        int sum = 0;
        Console.WriteLine("Tasodifiy sonlar:");
        for (int i = 0; i < randomNumbers.Length; i++)
        {
            randomNumbers[i] = random.Next(1, 101); // 1 dan 100 gacha
            Console.Write(randomNumbers[i] + " ");
            sum += randomNumbers[i];
        }
        double average = (double)sum / randomNumbers.Length;
        Console.WriteLine("\nSonlar yig'indisi: " + sum);
        Console.WriteLine("O'rtacha qiymat: " + average);
    }
}
```

Ushbu maqolada tasodifiy sonlarni generatsiya qilish algoritmilari va ularning dasturlashda qo'llanilishi ko'rib chiqildi. Tasodifiy sonlar ko'plab sohalarda, jumladan, simulyatsiya, statistik tahlil va kriptografiyada qo'llaniladi. C# dasturlash tili tasodifiy sonlar yaratishda qulay va keng imkoniyatlarni taqdim etadi. Ushbu masala va algoritmalar dasturiy yechimlarni takomillashtirishga yordam beradi.





C# dasturlash tilida tasodifiy sonlarga oid algoritmlar ko'plab ilovalarda qo'llanilishi mumkin. Bu metodlar orqali tasodifiy butun sonlar, haqiqiy sonlar, boolean qiymatlar va boshqa turdagi tasodifiy ma'lumotlarni yaratish mumkin. Bundan tashqari, tasodifiy sonlar ko'plab ilg'or algoritmlar va simulyatsiyalar uchun asosiy vosita hisoblanadi. C# dasturlash tili taqdim etadigan Random sinfi yordamida bu jarayonlarni sodda va samarali tarzda amalga oshirish mumkin.

Ushbu maqola C# dasturlash tilida tasodifiy sonlarni yaratish va ishlatishga oid asosiy tushunchalarni qamrab oldi. Tasodifiy sonlarni ishlab chiqish va qo'llashda ehtiyotkorlik zarur, chunki ular ko'plab muhim hisoblash va simulyatsiya jarayonlarida ishtirok etadi.

Foydalanilgan adabiyotlar:

1. Marcin Jamro. C# Data Structures and Algorithms. Second Edition. Published by Packt Publishing Ltd., in Birmingham, UK. 2024. – 349 p.
2. Дж.Эриксон. Алгоритмы.: – М.: " ДМК Пресс ", 2023. – 528 с.
3. Hemant Jain. Data Structures & Algorithms using Kotlin. Second Edition. in India. 2022. – 572 p.
4. Н. А. Тюкачев, В. Г. Хлебостроев. C#. Алгоритмы и структуры данных: учебное пособие для СПО. – СПб.: Лань, 2021. – 232 с.
5. Mykel J. Kochenderfer. Tim A. Wheeler. Algorithms for Optimization. Published by The MIT Press., in London, England. 2019. – 500 p.
6. Рафгарден Тим. Совершенный алгоритм. Графовые алгоритмы и структуры данных. – СПб.: Питер, 2019. - 256 с.
7. Ахо Альфред В., Ульман Джеффри Д., Хопкрофт Джон Э.
1. Структуры данных и алгоритмы. – М.: Вильямс, 2018. – 400 с.
8. Дж.Хайнеман, Г.Поллис, С.Стэнли. Алгоритмы. Справочник с примерами на C, C++, Java и Python, 2-е изд.: Пер. с англ. — СПб.: ООО "Альфа-книга", 2017. — 432 с.
9. Farmonov, S., & Nazirov, A. (2023). C# DASTURLASH TILIDA GRAY KODI BILAN ISHLASH. В CENTRAL ASIAN JOURNAL OF EDUCATION AND INNOVATION (Т. 2, Выпуск 12, сс. 71-74). Zenodo.
10. Farmonov, S., & Toirov, S. (2023). NETDA DASTURLASHNING ZAMONAVIY TEXNOLOGIYALARINI O'RGANISH. Theoretical aspects in the formation of pedagogical sciences, 2(22), 90-96
11. Raxmonjonovich, F. S. (2023). Array ma'lumotlar tizimini talabalarga o'qitishda Blockchain metodidan foydalanish. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 541-547.





12. Raxmonjonovich, F. S. (2023). Dasturlashda interfeyslardan foydalanishning ahamiyati. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 425-429.
13. Raxmonjonovich, F. S. (2023). Dasturlashda obyektga yo'naltirilgan dasturlashning ahamiyati. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 434-438.
14. Raxmonjonovich, F. S. (2023). Dasturlash tillarida fayllar bilan ishlash mavzusini Blended Learning metodi yordamida o'qitish. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 464-469.
15. Raxmonjonovich, F. S. (2023). DASTURLASHDA ISTISNOLARNING AHAMIYATI. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 475-481.
16. Raxmonjonovich, F. S. (2023). Dasturlashda abstraksiyaning o'rni. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 482-486.
17. Raxmonjonovich, F. S., & Ravshanbek o'g'li, A. A. (2023). Zamonaviy dasturlash tillarining qiyosiy tahlili. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 430-433.
18. Thomas, H. Algorithms Unlocked. MIT Press, 2013.
19. Heineman, G., Pollice, G. Algorithms in a Nutshell. O'Reilly Media, 2016.
20. Sedgewick, R., Wayne, K. Algorithms (4th Edition). Addison-Wesley, 2011.

