



C# DASTURLASH TILIDA GRAY KODI BILAN ISHLASH

Farmonov Sherzodbek Raxmonjonovich

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

Nazirov A'zamjon Azizjon o'g'li

Farg'ona davlat universiteti 2-kurs talabasi
azamjonnazirov000@gmail.com

<https://doi.org/10.5281/zenodo.10417073>

ARTICLE INFO

Qabul qilindi: 10-December 2023 yil

Ma'qullandi: 15- December 2023 yil

Nashr qilindi: 21- December 2023 yil

KEY WORDS

C#, Gray kodi, Elektron shovqinlar, Pozitsion sensorlar, Raqamli aloqa, Binar kod, Bitaralik kod, Binar format, Datchiklar, Xatolarni aniqlash va tuzatish.

ABSTRACT

Ushbu maqolada Gray kodi ning axborot texnologiyalari va raqamli tizimlardagi ahamiyati hamda uning qo'llanilishi ko'rib chiqilgan. Maqolada Gray kodi nima ekanligi, uning afzalliklari, raqamli signallarni qayta ishlashdagi va xatolarni aniqlashdagi roli hamda C# dasturlash tilida Gray kodini qo'llash bo'yicha amaliy masalalar hal etilgan. Shuningdek, Gray kodini dasturlashda qanday qo'llash mumkinligi, uning binomial kodlarga nisbatan afzalliklari va kamchiliklari muhokama qilingan

Bugungi kunda axborot texnologiyalari va raqamli tizimlarning tezkor rivojlanishi, bizning turmush tarzimizga chuqur singib, har bir sohani, jumladan sanoat, kommunikatsiya va ma'lumotlar qayta ishlashni o'zgartirmoqda. Ayni shu jarayonlarning muhim qismlaridan biri raqamli signallarni qayta ishlashdir. Raqamli tizimlar, ayniqsa, o'zgaruvchan signal qiymatlari bilan ishlaydigan qurilmalarda, masalan, datchiklarda yoki raqamli aloqa tizimlarida, xatolarni aniqlash va ularni minimal darajada ushlab turish uchun juda aniq va ishonchli kodlash usullariga muhtoj. Shu maqsadda ishlab chiqilgan kodlardan biri Gray kodi hisoblanadi.

Gray kodi qanday ishlaydi?

Har bir yangi son uchun, Gray kodida faqat bitta bit o'zgaradi. Bu "bir bit o'zgarish qoidasi" deyiladi va u raqamli tizimlarda xatolarni aniqlashni osonlashtiradi.

Misol uchun, agar siz 0 dan 15 gacha bo'lgan sonlarni Gray kodiga aylantirishni istasangiz, har bir keyingi son avvalgisidan faqat bitta bitning o'zgarishi bilan farqlanadi.

Gray kodida, ikkita ketma-ket sonning binar kodlari o'rtasida bir nechta bitlarning o'zgarishi mumkin bo'lgan bir paytda, Gray kodida faqat bitta bit o'zgaradi. Bu, raqamli tizimlarda xatolarni aniqlash va tuzatishni yengillashtiradi.

Gray kodining afzalliklari.

1.Xatolarni aniqlash va tuzatish: Raqamli tizimlar uchun Gray kodi yordamida xatolarni aniqlash va tuzatish jarayoni osonroq bo'ladi, chunki faqat bitta bitning o'zgarishi kutiladi.

2.Elektron shovqinlarga qarshilik: Gray kodi shovqinli muhitda ham ishonchli ishlashini

ta'minlaydi, chunki bitlarning o'zgarishi kamroq bo'ladi va shovqin tufayli noto'g'ri o'qilish ehtimoli kamayadi.

3.Pozitsion sensorlar: Pozitsiyani aniqlash tizimlarida (masalan, o'zgaruvchan induktiv datchiklar) Gray kodi keng qo'llaniladi, chunki u yanada aniq pozitsiyani ta'minlaydi.

4.Raqamli aloqa: Raqamli aloqa tizimlarida Gray kodidan foydalanish signalning buzilishiga chidamlilikni oshiradi va ma'lumotlarni yanada ishonchli uzatishni ta'minlaydi.

C# dasturlash tilida Gray kodining qo'llanilishi. C# kabi zamonaviy dasturlash tillarida Gray kodini qo'llash orqali dasturchilar turli xil raqamli tizimlar va qurilmalar uchun samarali va xatolarga chidamli dasturiy yechimlar yaratishlari mumkin. Gray kodini dasturlashda qo'llash uchun C# tilida yozilgan dastur sizga o'nlik sonni binar kodga aylantiradi va keyin bu binar kodni Gray kodiga aylantirish uchun qo'shimcha algoritmi qo'llashingiz kerak bo'ladi. Bu amaliyot ketma-ket bitlarni solishtirish orqali amalga oshiriladi.

Tahlil. Ushbu qismda, C# dasturlash tili yordamida Gray kodini qanday yaratish va ishlatish mumkinligini ko'rib chiqamiz. Quyida keltirilgan kod namunasi orqali Gray kodini hisoblash jarayoni tushuntiriladi.

```
using System;
namespace Graykodi
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("O'nlik sonning qiymatini kiriting: ");
            int n;
            while (!int.TryParse(Console.ReadLine(), out n))
            {
                Console.WriteLine("Noto'g'ri kiritildi! Iltimos, yana bir marta o'nlik son kiriting:");
            }

            string binaryCode = Convert.ToString(n, 2).PadLeft(4, '0');
            Console.WriteLine("Bitaralik kod: " + binaryCode);

            string grayCode = ConvertToGrayCode(binaryCode);
            Console.WriteLine("Gray kodi: " + grayCode);
            Console.ReadKey();
        }

        static string ConvertToGrayCode(string binary)
        {
            string gray = "";
            gray += binary[0];

            for (int i = 1; i < binary.Length; i++)
```

```
{  
    gray += (binary[i - 1] == binary[i] ? '0' : '1');  
}  
  
return gray;  
}  
}
```

Kodning Tahlili: Yuqoridagi kod, o'nlik sonni qabul qilib, uni avval binar kodga, keyin esa Gray kodiga aylantiradi. **Convert.ToString(n, 2).PadLeft(4, '0')**: Bu qator orqali o'nlik son binar formatga aylantiriladi va agar kerak bo'lsa, uning oldiga 0 qo'shiladi, bu esa har doim 4 xonali binar sonni kafolatlaydi. **ConvertToGrayCode funksiyasi:** Bu funksiya binar kodni Gray kodiga aylantirish uchun ishlatiladi. Gray kodi birinchi bitni o'zgartirmasdan olib, keyingi har bir bitni oldingi bit bilan ekskluziv yoki (XOR) amali orqali hisoblaydi. **gray += (binary[i - 1] == binary[i] ? '0' : '1')**: Bu qator orqali Gray kodining har bir biti hisoblanadi. Agar ketma-ket bitlar bir xil bo'lsa, Gray kodida 0 qo'shiladi, aks holda 1 qo'shiladi. Xulosa qilib aytganda, bu kod namunasi orqali, C# dasturlash tilida Gray kodi bilan ishlashning oddiy usuli ko'rsatilgan. Gray kodi raqamli tizimlarda signalning buzilishiga qarshi turish va xatolarni aniqlashda juda foydali bo'lishi mumkin. Raqamli tizimlar sohasida ishlaydigan mutaxassislar va dasturchilarga Gray kodini tushunish va uni qo'llash imkoniyatlari kengayadi.

Xulosa. Gray kodi, raqamli tizimlarda xatolarni aniqlash va tuzatish imkoniyatlarini yaxshilash, shuningdek, raqamli signallarni qayta ishlashni optimallashtirishda muhim rol o'ynaydi. C# tilida Gray kodini qo'llash, dasturchilarga turli xil raqamli tizimlar va qurilmalar uchun yanada samarali va ishonchli dasturiy echimlar yaratish imkonini beradi. Gray kodi va uning C# dasturlash tilidagi qo'llanilishi, axborot texnologiyalari sohasida yangi imkoniyatlar ochib beradi va raqamli tizimlarning samaradorligini oshirishda yordam beradi.

Foydalanilgan adabiyotlar ro'yhati:

1. John Paul Mueller. C# 10.0 All-in-One For Dummies. Published by: John Wiley & Sons, Inc., 2022. — 830 c.
2. A.Troelsen, R.Japikse. Pro C# 8 with .NET Core 3. Foundational Principles and Practices in Programming. Ninth Edition. Apress, 2020. – 1223 c.
3. A.Васильев. Программирование на C# для начинающих. Основные свидания.: – М.: "Эксмо", 2018. – 592 с.
4. Генри Уоррен. Алгоритмические трюки для програмистов. 2-е издание. — М. : Издательский дом "Вильямс", 2014. — 512 с.
5. Фармонов, Ш., & Камбарова, Д. (2022). КАК ПОМОЧЬ УЧЕНИКАМ РАЗВИТЬ ИНТЕРЕС К УЧЕБЕ. Центральноазиатский журнал образования и инноваций, 1(2), 118-120.
6. Farmonov, S., & Rahmatjonov, M. (2023). FLUTTER YORDAMIDA PLATFORMALARARO KUTUBXONANI ISHLAB SHIQISH TEXNOLOGIYASI. Центральноазиатский журнал образования и инноваций, 2(6 Part 2), 124-127.
7. Raxmonjonovich, F. S. (2023). USE OF BLENDED LEARNING TECHNOLOGY IN ORGANIZING INDEPENDENT EDUCATION OF STUDENTS. MATERIALLAR TO'PLAMI, 352.
8. Farmonov, S., & Karimova, M. (2023). MODERN METHODS TO DEVELOP MATHEMATICAL

THINKING IN SCHOOLCHILDREN. Бюллетень педагогов нового Узбекистана, 1(6 Part 2), 28-38.

9. Tojiyev, T., Boynazarov, A., & Farmonov, S. (2022). PHARMACOKINETICS IS A DESCRIPTION OF DRUGS AND THEIR BEHAVIOR IN THE HUMAN BODY BY BUILDING A MATHEMATICAL MODEL. Евразийский журнал медицинских и естественных наук, 2(13), 146-149.

10. Фармонов, Ш., & Хайдарова, С. (2022). Обобщенный метод Бубнова-Галеркина для уравнений с дробно-дифференциальным оператором. Norwegian Journal of Development of the International Science, (99), 10-15.

