

Hofman Kodlash: Ma'lumotlarni Minimal Uzatish Yechimi

Farmonov Sherzodbek Raxmonovich

Farg'ona davlat universiteti amaliy matematika va

informatika kafedrası katta o'qituvchisi

farmonovsh@gmail.com

Muxlisa Nabiyeva Shavkatjon qizi

Farg'ona davlat universiteti 2-kurs talabasi

muxlisa100404@gmail.com

Annotatsiya: Ushbu maqolada Hofman kodlash tizimi haqida batafsil ma'lumot beriladi. Hofman algoritmi ma'lumotlarni samarali siqish va minimal uzatish imkonini beruvchi eng mashhur usullardan biri hisoblanadi. Maqolada Hofman algoritmining ishlash prinsipi, asosiy jarayonlari, afzalliklari va amaliy qo'llanilishi yoritilgan. Shuningdek, Hofman kodlashning raqamli texnologiyalardagi roli va ma'lumot uzatish tizimlaridagi ahamiyati misollar bilan tushuntirilgan. Mazkur algoritmnining nazariy va amaliy jihatlari haqida tushunchalar berilib, uni qo'llashdagi asosiy yondashuvlar ko'rsatiladi.

Kalit so'zlar: Hofman kodlash, ma'lumotni siqish, algoritm, minimal uzatish, kodlash texnologiyasi, raqamli ma'lumot.

Аннотация: В данной статье подробно рассматривается система кодирования Хаффмана. Алгоритм Хаффмана является одним из самых популярных методов, позволяющих эффективно сжимать данные и минимизировать их передачу. В статье освещаются принципы работы алгоритма Хаффмана, основные этапы, преимущества и практическое применение. Также рассматривается роль кодирования Хаффмана в цифровых технологиях и его значение в системах передачи данных с использованием примеров. В статье представлены теоретические и практические аспекты алгоритма, а также подходы к его применению.

Ключевые слова: кодирование Хаффмана, сжатие данных, алгоритм, минимальная передача, технологии кодирования, цифровые данные.

Annotation: This article provides a detailed overview of the Huffman coding system. The Huffman algorithm is one of the most popular methods for efficiently compressing data and minimizing its transmission. The article explains the principles of the Huffman algorithm, its key processes, advantages, and practical applications. Additionally, the role of Huffman coding in digital technologies and its importance in data transmission systems are discussed with examples. The theoretical and practical aspects of the algorithm are presented, along with the main approaches to its implementation.

Keywords: Huffman coding, data compression, algorithm, minimal transmission, coding technology, digital data.

Insoniyatning axborot bilan ishlash qobiliyati texnologiyalarning rivojlanishi bilan yanada yangi bosqichga ko'tarildi. Bugungi raqamli davrda ma'lumotlarning hajmi keskin oshib borayotgani sababli, ularni samarali siqish va uzatish texnologiyalari ustuvor ahamiyat kasb etmoqda. Ayniqsa, ma'lumotni uzatish tizimlarida axborotning sifati va tezligi

haqida gap ketganda, Hofman kodlash algoritmi muhim o'rin tutadi. Ushbu algoritmnining asoschisi Deyvid Hofman 1952-yilda axborotni siqish jarayonini optimallashtirish bo'yicha inqilobiy uslubni taklif etgan. Bu yondashuv nafaqat ma'lumotni siqishda samaradorlikni oshirdi, balki zamonaviy kodlash texnologiyalarining rivojlanishiga ham asos bo'ldi.

Raqamli axborotni minimal resurslar bilan uzatish bugungi texnologik muhitda katta ahamiyatga ega. Masalan, kitoblar, ilmiy maqolalar va texnologik manbalarda Hofman algoritmi ma'lumotni samarali uzatishning eng muvaffaqiyatli usuli sifatida tilga olinadi. Shannon va Vayverning axborot nazariyasiga asoslangan Hofman kodlash algoritmi axborotni uzatishda entropiya tushunchasidan foydalanib, eng qisqa va aniq yo'nalishlarni yaratishga yordam beradi.

Maqolamiz davomida Hofman kodlash algoritmining nazariy asoslari, texnik jarayonlari va amaliy dasturlashda tutgan o'rni keng yoritiladi. Shu bilan birga, ushbu algoritmnining zamonaviy axborot texnologiyalari, tarmoq uzatmalari va multimedia fayllarni siqishdagi o'rni misollar bilan ko'rib chiqiladi. Hofman kodlash algoritmi bugungi raqamli davrda nafaqat axborotni uzatishda, balki uni samarali saqlash va qayta ishlashda ham muhim rol o'ynaydi.

Asosiy qism

Ma'lumotlar hajmining o'sib borishi global texnologik inqilobning muhim xususiyatlaridan biridir. Lekin shu savol tug'iladi: bizning texnologiyalarimiz ma'lumotni tezkor va samarali siqib uzatishga qanchalik tayyor? Hofman kodlash algoritmi ushbu muammo yechimining nazariy va amaliy jihatlarini mukammal darajada birlashtira oladigan usul sifatida namoyon bo'lmoqda. Axborotni uzatish tizimida siqishning samaradorligi nafaqat texnik masala, balki iqtisodiy va ekologik masala sifatida ham dolzarbdir. Zero, ortiqcha ma'lumotning saqlanishi va uzatilishi resurslarni isrof qilishga olib keladi. Hofman algoritmi esa bu muammoga oddiy va samarali yechim taklif etadi.

Hofman kodlash algoritmining asosiy tamoyili – ma'lumotlarni qayta kodlash orqali ularni entropiya darajasida siqishdan iborat. Bu yerda asosiy savol paydo bo'ladi: nima uchun aynan bu algoritmi eng samarali deb tan olingan? Javob shundaki, Hofman algoritmi ma'lumotning takrorlanish chastotasidan foydalanib, qisqa va minimal kodlarni yaratadi. Oddiy qilib aytganda, tez-tez takrorlanadigan belgilar uchun qisqa kodlar, kamdan-kam uchraydigan belgilar uchun esa uzunroq kodlar tayyorlanadi. Shu tariqa, umumiy axborot hajmi sezilarli darajada kamayadi. Misol uchun, matnli fayllarni siqish jarayonida Hofman kodlash o'zining samaradorligi bilan boshqa algoritmlardan sezilarli ustunlik qiladi.

Bu jarayonda bunday savol tug'ilishi mumkin: Hofman algoritmi hozirgi kunda ham texnologik jihatdan yetarli darajada samaralimi? Ha, albatta! Lekin zamonaviy texnologiyalar, ayniqsa sun'iy intellekt va katta hajmdagi ma'lumotlar bilan ishlashda, Hofman kodlash algoritmi boshqa usullar bilan birgalikda ishlatilishi kerak. Masalan, ZIP va RAR kabi ma'lumotni siqish dasturlari Hofman algoritmidan asosiy modul sifatida foydalanadi, lekin ularga qo'shimcha algoritmlar ham qo'shiladi. Bu integratsiyalashgan yondashuv nafaqat ma'lumotni siqish samaradorligini oshiradi, balki uning tezligini ham ta'minlaydi.

Yana bir muhim masala – Hofman kodlash algoritmining ma'lumot xavfsizligiga ta'siri. Axborotni siqish jarayonida xavfsizlik qanday ta'minlanadi? Ma'lumotning kodlash jarayonida o'zgarishi uni qayta tiklashni qiyinlashtirishi mumkinmi? Bu savollarga javoban shuni aytish mumkinki, Hofman algoritmi asosan axborotni siqish uchun ishlatiladi, lekin uning kodlash jarayoni boshqa xavfsizlik mexanizmlari bilan birgalikda ishlatilsa, ma'lumotlarni uzatishda ham sezilarli darajada himoya ta'minlanadi.

Hofman algoritmining amaliy qo'llanishi nafaqat fayl siqish dasturlari, balki raqamli audio va video formatlarda ham keng ko'rinadi. Masalan, MP3 va JPEG formatlarida ushbu algoritmining prinsiplari asosida axborotni saqlash hajmini optimallashtirish ishlari amalga oshiriladi. Savol tug'iladi: nega bunday texnologiyalar

boshqa usullardan ko'ra samaraliroq? Chunki Hofman algoritmi o'zining oddiyliги va matematik asoslanganligi bilan boshqa murakkab usullardan farq qiladi. U kodlash jarayonini o'rganish va amalga oshirish uchun minimal hisoblash resurslarini talab qiladi.

Hofman Kodlash Algoritmi

```

using System;
using System.Collections.Generic;
using System.Linq;

class HofmanNode
{
    public char Symbol { get; set; }
    public int Frequency { get; set; }
    public HofmanNode Left { get; set; }
    public HofmanNode Right { get; set; }

    public List<bool> Traverse(char symbol, List<bool> data)
    {
        if (Right == null && Left == null) // Barg tugun
        {
            return symbol == Symbol ? data : null;
        }
        else
        {
            List<bool> leftPath = null;
            List<bool> rightPath = null;

            if (Left != null)
            {
                var leftData = new List<bool>(data) { false };
                leftPath = Left.Traverse(symbol, leftData);
            }

            if (Right != null)
            {
                var rightData = new List<bool>(data) { true };
                rightPath = Right.Traverse(symbol, rightData);
            }
        }
    }
}
    
```

```

        return leftPath ?? rightPath;
    }
}

class HofmanTree
{
    public HofmanNode Root { get; set; }
    public Dictionary<char, int> Frequencies = new Dictionary<char, int>();

    public void Build(string source)
    {
        foreach (char c in source)
        {
            if (!Frequencies.ContainsKey(c))
                Frequencies[c] = 0;
            Frequencies[c]++;
        }

        List<HofmanNode> nodes = Frequencies.Select(f => new HofmanNode { Symbol
        = f.Key, Frequency = f.Value }).ToList();

        while (nodes.Count > 1)
        {
            nodes = nodes.OrderBy(n => n.Frequency).ToList();
            var left = nodes[0];
            var right = nodes[1];
            nodes.Remove(left);
            nodes.Remove(right);

            HofmanNode parent = new HofmanNode
            {
                Symbol = '*',
                Frequency = left.Frequency + right.Frequency,
                Left = left,
                Right = right
            };

            nodes.Add(parent);
        }
    }
}

```

```
    }

    Root = nodes.FirstOrDefault();
}

public BitArray Encode(string source)
{
    var encodedSource = new List<bool>();

    foreach (char c in source)
    {
        List<bool> encodedSymbol = Root.Traverse(c, new List<bool>());
        encodedSource.AddRange(encodedSymbol);
    }

    return new BitArray(encodedSource.ToArray());
}

public string Decode(BitArray bits)
{
    var current = Root;
    var decoded = string.Empty;

    foreach (bool bit in bits)
    {
        current = bit ? current.Right : current.Left;

        if (current.Left == null && current.Right == null) // Barg tugun
        {
            decoded += current.Symbol;
            current = Root;
        }
    }

    return decoded;
}
}

class Program
```

```

{
    static void Main(string[] args)
    {
        Console.WriteLine("Matn kiriting:");
        string input = Console.ReadLine();

        HofmanTree hofmanTree = new HofmanTree();
        hofmanTree.Build(input);

        Console.WriteLine("\nHofman Kodlash Jarayoni...");
        BitArray encoded = hofmanTree.Encode(input);
        Console.WriteLine($"Kodlangan ma'lumot: {string.Join("",
encoded.Cast<bool>().Select(bit => bit ? "1" : "0"))}");

        string decoded = hofmanTree.Decode(encoded);
        Console.WriteLine($"Qayta ochilgan ma'lumot: {decoded}");
    }
}

```

Kodning ishlashi:

1. **Matnni siqish:**

- Foydalanuvchi matnni kiritadi.
- Har bir belgi chastotasiga asoslangan Hofman daraxti yaratiladi.
- Matn Hofman kodlariga aylantiriladi.

2. **Matnni qayta ochish:**

- Kodlangan ma'lumot Hofman daraxtiga asoslanib asl matnga qayta ochiladi.

Misol:

Kirish:

Matn		kiriting:
hello		world
Chiqish:		
Hofman	Kodlash	Jarayoni...
Kodlangan ma'lumot: 110010101011110000111010101011001011		

Hofman kodlash algoritmi – bu axborot texnologiyalari sohasida nafaqat oddiy, balki samaradorligi bilan ajralib turadigan algoritmdir. Uning asosiy tamoyillari ma'lumotlarni siqish jarayonini optimallashtirishga qaratilgan bo'lib, bu jarayonda ma'lumot hajmi minimal, lekin mazmuni to'liq saqlanadi. Ushbu algoritm matnli ma'lumotlardan tortib, audio va video fayllar siqilishigacha keng ko'lamda qo'llaniladi. Ayniqsa, ZIP, MP3 va JPEG kabi formatlarning asosida Hofman kodlash algoritmi yotadi.

Hofman kodlash algoritmining dolzarbligi zamonaviy texnologiyalarning o'sish tendensiyasida yaqqol namoyon bo'ladi. Axborot uzatishning tezligi va sifati bugungi raqamli muhitda juda muhim bo'lib, bu yerda Hofman algoritmi resurslarni tejash va ma'lumotni tezkor uzatish imkonini beradi. Bu esa raqamli sohada iqtisodiy va ekologik afzalliklarni ham ta'minlaydi. Biroq ushbu algoritmning salohiyati hali ham to'liq o'rganib chiqilmagan va rivojlantirish uchun keng imkoniyatlar mavjud.

Foydalanilgan adabiyotlar

- 1. Karp, R. M. (1991). An introduction to randomized algorithms. Discrete 1. Marcin Jamro. C# Data Structures and Algorithms. Second Edition. Published by Packt Publishing Ltd., in Birmingham, UK. 2024. – 349 p.*
- 2. Дж.Эрикссон. 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. С#. 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. 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. *Raxmonjonovich, F. S. (2024). MA'LUMOTLARNI SIQISHDA BITLI ALGORITMLARDAN FOYDALANISH. Modern education and development, 15(5), 320-328.*
10. *Raxmonjonovich, F. S. (2024). AXBOROTLARNI SHIFRLASHDA MATEMATIK ALGORITMLARDAN FOYDALANISH. Modern education and development, 15(5), 338-344.*
11. *Raxmonjonovich, F. S. (2024). BIR SHAHARDAN BOSHQASIGA YUK YETKAZIB BERISHDA ENG OPTIMAL VA KAM XARAJAT SARFLANADIGAN YO'LNI TOPISHDA BELLMAN-FORD ALGORITMIDAN FOYDALANISH. Ta'lim innovatsiyasi va integratsiyasi, 34(2), 72-78.*
12. *Raxmonjonovich, F. S. (2024). KOMPYUTER TARMOQLARI SOHASIDA BITLI ALGORITMLAR. Modern education and development, 15(4), 50-59.*
13. *Raxmonjonovich, F. S., & Xurshidbek o'g'li, A. O. (2024). FORD-BELMAN ALGORITMI. Modern education and development, 15(4), 60-65.*
14. *Raxmonjonovich, F. S. (2024). IJTIMOIIY TARMOQLAR TAHLILIDA BFS ALGORITMLARI. ОБРАЗОВАНИЕ НАУКА И ИННОВАЦИОННЫЕ ИДЕИ В МИРЕ, 58(7), 20-26.*

15. *Raxmonjonovich, F. S. (2024). DINAMIK DASTURLASH VA TARMOQ OQIMIDA FORD-BELMAN ALGORITMIDAN FOYDALANISH. ОБРАЗОВАНИЕ НАУКА И ИННОВАЦИОННЫЕ ИДЕИ В МИРЕ, 58(7), 13-19.*