Muminov Dilmurod Yunus o‘g‘li

Urganch davlat universiteti Matematik injiniring yo‘nalishi talabasi

C# DASTURLASH MUHITIDA IKKILIK SANOQ SISTEMASIDAGI HAQIQIY SONLAR USTIDA ARIFMETIK AMALLARNI BAJARADIGAN DASTUR TUZISH.

Annotatsiya. Ushbu maqolada C# dasturlash muhitida ikkilik sistemasidagi haqiqiy sonlar ustida arifmetik amallarni bajaradigan dastur tuzish masalasi ko‘rib chiqilgan.

Аннотация. В данной статье рассматривается вопрос создания программы, выполняющей арифметические операции над действительными числами в двоичной системе в среде программирования C#.

Annotation. This article discusses how to create a program in the C# programming environment that performs arithmetic operations on real numbers in binary.

Kalit so‘zlar: Algoritm, Takrolanish operatorlari, Metodlar.

Ключевые слова: алгоритм, операторы повторения, методы.

Key words: Algorithm, Repetition operators, Methods.

Dastlab bir sanoq sistemasidan ikkinchi sanoq sistemasiga o‘tish masalasini qarasak. Ma’lumki o‘nlik sanoq sistemasidagi sonlarni ikkilik sanoq sistemasiga (umuman, boshqa sanoq sistemalariga, masalan, sakkizlik, o‘n oltilik va h.k.) o‘tkazish uchun berilgan son o‘tkaziladigan sanoq sistema asosiga natija birga teng bo‘lguncha ketma-ket bo‘linadi va qoldiqlar o‘ngdan chapga qarab yoziladi. Agar o‘nlik sanoq sistemasidagi sonning kasr qismi mavjud bo‘lsa, ushbu sonning butun qismi alohida, kasr qismi alohida boshqa sanoq sistemasiga o‘tiladi.Kasr qismni boshqa sanoq sistemasiga o‘tkazish uchun quyidagi qoidadan foydalanamiz:

O‘nlik sanoq sistemasidagi kasr sonni ikkilik (sakkizlik, o‘n oltilik yoki boshqa) sanoq sistemasiga o‘tkazish uchun shu sonni va keyingi hosil bo‘lgan kasr qismlarni o‘tilayotgan sanoq sistemaning asosiga, ya’ni ikkiga (sakkizlik sanoq sistemasida sakkizga, o‘n oltilik sanoq sistemasida o‘n oltiga) ketma-ket ko‘paytirish lozim. Natijada hosil bo‘lgan butun qismlarni to‘g‘ri tartibda yozib chiqish kerak. Ko‘paytirish jarayoni ikki holda to‘xtashi mumkin: 1) kasr qismida nollar hosil bo‘lgan taqdirda; 2) berilgan razryad aniqlikka erishilgan holda.

Ikkilik, sakkizlik, o‘n oltilik va boshqa sanoq sistemalaridan o‘nlik sanoq sistemalariga o‘tish bir xil, ya’ni berilgan asos darajalari bo‘yicha yoyish qoidasi bo‘yicha o‘tiladi va bu qoida quyidagicha:

Ushbu son k xonali, n asosli (n lik sanoq sistemasidagi son) $a_1a_2\ldots a_{kn}$ bo‘lsin (k ta raqam bor). U holda $a_1a_2\ldots a_{kn}=a_1*n^{k-1}+ a_2*n^{k-2}+ a_3*n^{k-3}+\ldots+ a_{k-1}*n^1+ a_k*n^0$ dan foydalanib n lik sanoq sistemasidan o‘nlik sanoq sistemasiga o‘tish mumkin.

Shu o‘rinda ta’kidlash joizki, yuqoridagi qoida haqiqiy sonlar uchun ham o‘rinli.Bunda sonning kasr qismi darajasi -1 dan boshlanadi.

Yuqoridagi qoidalarga asosan dasturning algoritmidagi amallarning bajarilish tartibi quyidagicha bo‘ladi:

-bajariladigan amal kiritiladi;

-sonlar kiritiladi;

-kiritilgan ikkilik sanoq sistemadagi sonlar o‘nlik sanoq sistemasiga o‘tkaziladi;

-o‘nlik sanoq sistemasiga o‘tkazilgan sonlar ustida kiritilgan amal bajariladi;

-hosil bo‘lgan natija o‘nlik sanoq sistemasidan ikkilik sanoq sistemasiga o‘tkaziladi;

-natija ekranga chiqariladi.

Ushbu algoritm asosida masalani yechishda uchta metod yaratamiz. Bunda bitta metodimiz asosiy va ikkita metodimiz yordamchi bo‘ladi. Asosiy metodimiz ikkilik sanoq sistemasidagi sonlarni o‘nlik sanoq sistemasiga o‘tkazishda, bu sonlar ustida kiritilgan amalni bajarishda va natijani ikkilik sanoq sistemasiga o‘tkazishda xizmat qiladi. Yordamchi metodlarimiz esa o‘z navbatida ikkilik sanoq sistemasidagi sonning kasr qismini o‘nlik sanoq sistemasiga o‘tkazishda va o‘nlik sanoq sistemasidagi sonning kasr qismini ikkilik sanoq sistemasiga o‘tkazishda yordamga keladi.

Yordamchi metodlarimizni ikkilik sanoq sistemasidagi sonlarning kasr qismini o‘nlik sanoq sistemasiga o‘tkazuvchi va o‘nlik sanoq sistemasidagi sonlarning kasr qismini ikkilik sanoq sistemasiga o‘tkazuvchi qilib yaratamiz.Bu metodlarimiz qiymat qaytaradigan bo‘lib, biri o‘nlik sanoq sistemasidagi kasr sonni ifodalovchi haqiqiy sonni, ikkinchisi ikkilik sanoq sistemasidagi kasr qismni o‘zida ifodalovchi satrni qaytaradigan bo‘ladi va biri parametrida satrni, ikkinchisi haqiqiy sonni (kasr son) qabul qiladi.

Quyida ushbu metodlarning C# dasturlash tilida kod holida yozilishini ko‘rib chiqamiz:

```
static double KasrIkkilikdanOnlikgaOtish(string kasrqism)
    {
        int raqam, daraja = -1;
        double kasr = 0;
        for (int i = 0; i < kasrqism.Length; i++)
        {
            raqam = int.Parse(kasrqism[i].ToString());
            kasr += raqam * Math.Pow(2, daraja);
            daraja--;
        }
        return kasr;
    }
```

Bu metodimizda for sikl operatori orqali yuqoridagi qoidaga asosan ikkilik sanoq sistemasidagi sonning kasr qismini o‘nlik sanoq sistemasiga o‘tkazishda xizmat qiladi. O‘nlik sanoq sistemasidagi sonning kasr qismini ikkilik sanoq sistemasiga o‘tkazuvchi metodimizda while shart takrorlanish operatori orqali sonning kasr qismini boshqa sanoq sistemasiga o‘tkazish qoidasi yordamida sonning kasr qismini ikkilik sanoq sistemasida ifodalaymiz, va ushbu metodimiz quyidagicha bo‘ladi:

```
static string KasrniIkkiikgaOtkazish(double kasrSon)
  {
    string a = "";

    while (kasrSon != 0)
    {
      a += Math.Floor(kasrSon * 2);
      kasrSon = kasrSon * 2 - Math.Floor(kasrSon * 2);
    }
    return a;

    }
```

Asosiy metodimiz qiymat qaytarmaydigan bo‘lib, uning ishlash prinsipi quyidagicha bo‘ladigan qilib tuzamiz:

-sonlar ikkilik sanoq sistemasiga o‘tkaziladi (bunda agar sonlarning kasr qismi mavjud bo‘lsa, kasr qismi alohida, butun qism alohida o‘nlik sanoq sistemasiga o‘tkaziladi);

-sonlar ustida kiritilgan amal bajariladi;

-natija ikkilik sanoq sistemasiga o‘tkaziladi;

-natija ekranga chiqariladi;

Shu o‘rinda ta’kidlash joizki, sonlarni manfiy bo’lgan holni, natija manfiy bo‘lgan hollarni e’tiborga olib, metodimizda barcha bo‘lishi mumkin bo‘lgan hollarga alohida e’tibor beramiz.Quyida ushbu metodning C# dasturlash tilida kod holida yozilishini ko‘rib chiqamiz:

```
static void Ikkilik(string son1, string son2, char amal)
    {
      //Birinchi sonni musbat yoki manfiyligini aniqlaymiz
        bool manfiy1 = false, manfiy2 = false, kasr1 = false, kasr2 = false;
        int k1 = 0, k2 = 0;
        if (son1[0] == '-')
        {
          k1++;
          manfiy1 = true;
        }
        //Ikkinchi sonni musbat yoki manfiyligini aniqlaymiz
        if (son2[0] == '-')
```

```
{
    k2++;
    manfiy2 = true;
}
    //berilgan sonlarni kasr va butun qismlarga ajratamiz
    bool t2 = false;
    string son1butun = "", son1kasr = "";
    for (int i = k1; i < son1.Length; i++)
    {
        if (son1[i] != ',')
        {
            son1butun += son1[i];
        }
        else
        {
            for (int j = i + 1; j < son1.Length; j++)
            {
                son1kasr += son1[j];
            }
            t2 = true;
        }
        if (t2)
        {
            break;
        }}
    t2 = false;
    string son2butun = "", son2kasr = "";
    for (int i = k2; i < son2.Length; i++)
    {
        if (son2[i] != ',')
        {
            son2butun += son2[i];
        }
        else
        {
            for (int j = i + 1; j < son2.Length; j++)
            {
                son2kasr += son2[j];
            }
            t2 = true;
        }
        if (t2)
        {
```

```
                    break;
                }
            }
            //sonlarni o'nlik sanoq sistemasiga o'tkazamiz
            int x = 0;
            double s1 = 0, s2 = 0, s = 0;
for (int i = 0, j= son1butun.Length - 1; i < son1butun.Length && j >= 0; i++, j--)
            {
                x = int.Parse(son1butun[i].ToString());
                s1 += x * Math.Pow(2, j);
            }
for (int i =0, j = son2butun.Length - 1; i < son2butun.Length && j >= 0; i++, j--)
            {
                x = int.Parse(son2butun[i].ToString());
                s2 += x * Math.Pow(2, j);
            }
 //sonning kasr qismi bor yo'qligini tekshirib, bo'lsa butun qismga qo'shamiz
            if (kasr1)
            {
                s1 += KasrIkkilikdanOnlikgaOtish(son1kasr);
            }
            if (kasr2)
            {
                s2 += KasrIkkilikdanOnlikgaOtish(son2kasr);
            }
            //sonlarni ishorasini qo'yamiz
            if (manfiy1)
            {
                s1 = -s1;
            }
            if (manfiy2)
            {
                s2 = -s2;
            }
            //sonlar ustida amal bajaramiz
            if (amal == '+')
            {
                s = s1 + s2;
            }
            if (amal == '-')
            {
                s = s1 - s2;
            }
```

```
if (amal == '*')
{
   s = s1 * s2;
}
if (amal == '/')
{
   s = s1 / s2;
}
//natijani ikkilik sistemaga o'tkazamiz
string b1 = "", b2 = ""; ;
int son = 0, y;
bool natijakasr = false;
double kasrqism = 0;
   try
   {
      son = int.Parse(s.ToString());
   }
   catch
   {
      if (s > 0)
      {
         son = int.Parse(Math.Floor(s).ToString());
         kasrqism = s - Math.Floor(s);
      }
      if (s < 0)
      {
         son = int.Parse(Math.Floor(s).ToString()) + 1;
         kasrqism = son - s;
      }
      natijakasr = true;
   }
   if (s < 0)
   {
      b2 += '-';
   }
   if (son < 0)
   {
      son = Math.Abs(son);
   }
   if (son == 0)
   {
      b2 += 0;
   }
```

```
        while (son > 0)
        {
            y = son % 2;
            son = son / 2;
            b1 += y;
        }
        for (int i = b1.Length - 1; i >= 0; i--)
        {
            b2 += b1[i];
        }
        if (natijakasr)
        {
            b2 += ',';
            b2 += KasrniIkkiikgaOtkazish(kasrqism);
        }
        Console.WriteLine("Natija: " + b2);
    }
```

Endi hosil qilingan metodlar yaxlit bir dastur holiga keltirilsa, ikkilik sanoq sistemasidagi ikkita haqiqiy son ustida matematik amallardan birini bajaruvchi dastur hosil bo‘ladi.

Foydalanilgan adabiyotlar :

1. U. Yaxshiyev ,,Informatikadan qisqacha ma’lumotlar ‘‘ Toshkent, 2019-yil.
2. I. Baltayeva, Y. Xudayberganov, N. Vaisova ,,C# dasturlash tiliga kirish ‘‘ Urganch, 2015-yil
3. С. Наков, В.Колев и колектив “Въведение в програмирането със C#” София, 2011 .
4. https://www.geeksforgeeks.org/convert-decimal-fraction-binary-number/ sayti.
5. https://www.tutorialspoint.com sayti