ASOSIY VOSITALAR AMORTIZATSIYASI HISOBINI 1C DASTURLASH TILIDA AVTOMATLASHTIRISH

Ibroximova Hayitxon Mirzoxidjon qizi

Andijon davlat texnika instituti

Axborot tizimlari va texnologiyalari yo‘nalishi 3-bosqich talabasi

Annotatsiya: Maqolada korxonalarda asosiy vositalar va ularning amortizatsiyasi hisobini 1C dasturlash tili va .NET platformasi asosida avtomatlashtirish masalalari yoritilgan. Amortizatsiyani hisoblashning uchta usuli (to‘g‘ri chiziqli, kamayuvchi qoldiq va ishlab chiqarish hajmiga mutanosib usul) dasturiy ifodalanib, ularning natijalari solishtirilgan. Ma’lumotlar bazasi tuzilmasi va buxgalteriya yozuvlarini avtomatik shakllantirish algoritmi taklif etilgan. Dasturiy yechim qo‘lda hisoblashdagi xatolarni kamaytirishi va hisobot tayyorlash vaqtini qisqartirishi asoslab berilgan.

Kalit so‘zlar: asosiy vositalar, amortizatsiya, buxgalteriya hisobi, C#, .NET, avtomatlashtirish, ma’lumotlar bazasi.

Аннотация: В статье рассматривается автоматизация учёта основных средств и начисления амортизации на предприятиях средствами языка C# и платформы .NET. Реализованы и сопоставлены три метода начисления амортизации: линейный, уменьшаемого остатка и пропорциональный объёму продукции. Предложены структура базы данных и алгоритм автоматического формирования бухгалтерских проводок.

Ключевые слова: основные средства, амортизация, бухгалтерский учёт, C#, .NET, автоматизация, база данных.

Abstract: The article examines the automation of fixed asset and depreciation accounting in enterprises using C# and the .NET platform. Three depreciation methods (straight-line, declining balance and units-of-production) are implemented in code and compared. A database structure and an algorithm for automatic generation of accounting entries are proposed. The solution reduces manual calculation errors and shortens reporting time.

Keywords: fixed assets, depreciation, accounting, C#, .NET, automation, database.

Kirish

Bugungi kunda O‘zbekiston korxonalarida raqamli iqtisodiyotni rivojlantirish buxgalteriya hisobini yuritish jarayonlarini avtomatlashtirishni taqozo etmoqda. Asosiy vositalar korxona mulkining muhim qismini tashkil etadi, ularning amortizatsiyasi esa xarajatlar tarkibiga, moliyaviy natijaga va soliq majburiyatlariga bevosita ta’sir ko‘rsatadi. Amortizatsiyani qo‘lda yoki elektron jadvallarda hisoblash katta hajmdagi ma’lumotlarda xatolarga, hisoblash usullarini almashtirishda murakkabliklarga olib keladi.

Tadqiqotning **maqsadi** asosiy vositalar amortizatsiyasini hisoblash va uni buxgalteriya yozuvlarida aks ettirishni C# tilida avtomatlashtirish modelini ishlab chiqishdan iborat. Maqsadga erishish uchun quyidagi **vazifalar** qo‘yildi.

Amortizatsiya hisoblash usullarining iqtisodiy mohiyatini tahlil qilish.

Ma’lumotlar bazasi tuzilmasini loyihalash.

Hisoblash algoritmlarini C# da amalga oshirish.

Usullar natijalarini solishtirib, amaliy xulosalar berish.

Tadqiqot obyekti sifatida korxonaning asosiy vositalar hisobi jarayoni, predmeti sifatida esa ushbu jarayonni avtomatlashtiruvchi dasturiy ta’minot olingan.

Asosiy vositalar hisobi O‘zbekiston Respublikasining «Buxgalteriya hisobi to‘g‘risida»gi Qonuni, 5-son «Asosiy vositalar» buxgalteriya hisobi milliy standarti (BHMS) hamda Soliq kodeksi bilan tartibga solinadi. Standartga ko‘ra, amortizatsiya asosiy vosita foydali xizmat muddati davomida uning amortizatsiya qilinadigan qiymatini muntazam taqsimlash orqali hisoblanadi. Ushbu tartib moliyaviy hisobot uchun hisoblangan amortizatsiya va soliq maqsadlaridagi amortizatsiya o‘rtasida farq paydo bo‘lishiga ham sabab bo‘ladi, shu sababli dasturda ikkala hisobni alohida yuritish maqsadga muvofiq.

Dasturiy ta’minot nuqtai nazaridan 1C tili obyektga yo‘naltirilgan dasturlash, kuchli tiplashtirish va Entity Framework Core kutubxonasi orqali ma’lumotlar bazasi bilan qulay ishlash imkonini beradi. Bu uni korxona darajasidagi buxgalteriya ilovalari uchun amaliy tanlovga aylantiradi.

Tadqiqotda tizimli tahlil, modellashtirish va solishtirma tahlil usullaridan foydalanildi. Dastur .NET platformasida, ma’lumotlar bazasi sifatida SQL Server, ma’lumotlar bilan ishlash uchun Entity Framework Core qo‘llanildi.

Tizim quyidagi asosiy jadvallardan iborat:

FixedAssets: inventar raqami, nomi, guruhi, boshlang‘ich qiymati, tugatish (qoldiq) qiymati, foydali xizmat muddati, foydalanishga topshirilgan sana, amortizatsiya usuli;

DepreciationEntries: aktiv identifikatori, davr (oy), hisoblangan summa, to‘plangan eskirish, qoldiq qiymat;

Accounts: schyotlar rejasi (masalan, asosiy vositalar, to‘plangan eskirish, davr xarajatlari schyotlari);

Journal: avtomatik shakllantirilgan buxgalteriya yozuvlari (debet, kredit, summa, sana).

Amortizatsiyani hisoblash usullari

1. To‘g‘ri chiziqli usul. Yillik amortizatsiya = (Boshlang‘ich qiymat − Qoldiq qiymat) / Foydali xizmat muddati.

2. Kamayuvchi qoldiq usuli. Yillik amortizatsiya = Balans qiymati × Me’yor, bunda me’yor ikkilangan to‘g‘ri chiziqli stavkaga teng olinadi. Oxirgi yilda amortizatsiya qoldiq qiymatdan pastga tushmaydigan miqdor bilan cheklanadi.

3. Ishlab chiqarish hajmiga mutanosib usul. Amortizatsiya = (Boshlang‘ich qiymat − Qoldiq qiymat) / Umumiy ish hajmi × Davrdagi ish hajmi.

Dasturiy amalga oshirish

Hisoblash mantiqi strategiya (Strategy) namunasi asosida yaratildi: har bir usul umumiy interfeysni amalga oshiruvchi alohida sinf bo‘ladi. Bu yangi usul qo‘shganda mavjud kodni o‘zgartirmaslikka imkon beradi.

```
public interface IDepreciationMethod
{
    decimal Calculate(FixedAsset asset, int year, decimal usage = 0);
}

public class StraightLine : IDepreciationMethod
{
    public decimal Calculate(FixedAsset a, int year, decimal usage = 0)
        => (a.Cost - a.SalvageValue) / a.UsefulLifeYears;
}

public class DecliningBalance : IDepreciationMethod
```

```
{
    public decimal Calculate(FixedAsset a, int year, decimal usage = 0)
    {
        decimal rate = 2m / a.UsefulLifeYears;
        decimal bookValue = a.Cost;
        for (int i = 1; i < year; i++)
            bookValue -= Math.Min(bookValue * rate, bookValue - a.SalvageValue);
        return Math.Min(bookValue * rate, bookValue - a.SalvageValue);
    }
}

public class UnitsOfProduction : IDepreciationMethod
{
    public decimal Calculate(FixedAsset a, int year, decimal usage = 0)
        => (a.Cost - a.SalvageValue) / a.TotalUnits * usage;
}
```

Oy yakunida tizim barcha faol aktivlar bo‘yicha amortizatsiyani hisoblaydi, DepreciationEntries jadvaliga yozadi va jurnalda avtomatik yozuv hosil qiladi: **Debet** (davr xarajatlari schyoti) / **Kredit** (to‘plangan eskirish schyoti). Har bir amal foydalanuvchi va vaqt belgisi bilan qayd etiladi, bu ichki audit imkonini beradi.

Natijalar va muhokama

Usullarni solishtirish uchun shartli misol olindi: asosiy vosita boshlang‘ich qiymati 120 000 000 so‘m, qoldiq qiymati 12 000 000 so‘m, foydali xizmat muddati 5 yil. Ishlab chiqarish usuli uchun yillik ish hajmi (mashina-soat) 20 000, 25 000, 20 000, 15 000 va 10 000 deb qabul qilindi (jami 90 000). Ma’lumotlar tasviriy xarakterga ega.

Yil	To‘g‘ri chiziqli	Kamayuvchi qoldiq	Ishlab chiqarish usuli
1	21 600 000	48 000 000	24 000 000
2	21 600 000	28 800 000	30 000 000
3	21 600 000	17 280 000	24 000 000
4	21 600 000	10 368 000	18 000 000
5	21 600 000	3 552 000	12 000 000
Jami	108 000 000	108 000 000	108 000 000

Jadvaldan ko‘rinadiki, usuldan qat’i nazar, amortizatsiya qilinadigan qiymat (108 000 000 so‘m) to‘liq taqsimlanadi, farq esa yillar bo‘yicha taqsimotda. Kamayuvchi qoldiq usulida birinchi yilda xarajat eng yuqori bo‘ladi (48 000 000 so‘m), bu tez eskiradigan texnika uchun mos keladi. To‘g‘ri chiziqli usul barqaror xarajatni ta’minlaydi va rejalashtirishni osonlashtiradi. Ishlab chiqarish usuli esa xarajatni haqiqiy foydalanish intensivligiga bog‘laydi.

Avtomatlashtirishning amaliy afzalliklari

Hisoblash xatolari va takroriy kiritishlar kamayadi.

Usulni o‘zgartirish va qayta hisoblash bir necha soniyada bajariladi.

Jurnal yozuvlari va hisobotlar (aktivlar qaydnomasi, eskirish qaydnomasi) avtomatik shakllanadi.

Amallar tarixi saqlanib, audit osonlashadi.

Dasturning arxitekturasi va modullari

Tizim uch qatlamli arxitekturada loyihalandi: ma’lumotlar qatlami (SQL Server va Entity Framework Core), biznes-mantiq qatlami (hisoblash va buxgalteriya yozuvlari xizmatlari) hamda foydalanuvchi interfeysi (WPF yoki Windows Forms). Qatlamlarning bunday ajratilishi hisoblash mantiqini interfeysdan mustaqil sinash va keyinchalik veb-ilovaga o‘tkazish imkonini beradi.

Asosiy modullar quyidagilardan iborat:

1. **Aktivlar reyestri**: asosiy vositalarni qabul qilish, inventar raqami berish, guruhlash, hisobdan chiqarish;

2. **Amortizatsiya moduli**: tanlangan usul bo‘yicha oylik hisoblash va davrni yopish;

3. **Jurnal moduli**: hisoblangan summalar bo‘yicha buxgalteriya yozuvlarini avtomatik shakllantirish;

4. **Hisobotlar moduli**: asosiy vositalar qaydnomasi, eskirish qaydnomasi va davr bo‘yicha aylanma qaydnomasi;

5. **Ma’muriy modul**: foydalanuvchilar, rollar va amallar jurnali (audit log).

Ma’lumotlar bazasi modelini Entity Framework Core orqali quyidagicha ifodalash mumkin:

```
public class FixedAsset
{
    public int Id { get; set; }
    public string InventoryNumber { get; set; } = "";
    public string Name { get; set; } = "";
    public decimal Cost { get; set; }
    public decimal SalvageValue { get; set; }
    public int UsefulLifeYears { get; set; }
    public decimal TotalUnits { get; set; }
    public DateTime CommissionDate { get; set; }
    public string Method { get; set; } = "StraightLine";
    public List<DepreciationEntry> Entries { get; set; } = new();
}

public class DepreciationEntry
```

```
{
    public int Id { get; set; }
    public int FixedAssetId { get; set; }
    public DateTime Period { get; set; }
    public decimal Amount { get; set; }
    public decimal AccumulatedDepreciation { get; set; }
    public decimal BookValue { get; set; }
}
```

Oylik hisoblash va jurnal yozuvini shakllantirish

Oylik amortizatsiya yillik summaning 12 ga bo‘linishi bilan topiladi. Hisoblash davri aktiv foydalanishga topshirilgan oydan keyingi oydan boshlanadi, hisobdan chiqarilgan aktiv uchun esa to‘xtatiladi. Quyidagi xizmat barcha faol aktivlar bo‘yicha davrni yopadi va har bir aktiv uchun hisoblangan summani saqlaydi:

```
public class DepreciationService
{
    private readonly AppDbContext _db;
    private readonly Dictionary<string, IDepreciationMethod> _methods;

    public DepreciationService(AppDbContext db,
        Dictionary<string, IDepreciationMethod> methods)
    {
        _db = db;
        _methods = methods;
    }

    public void ClosePeriod(DateTime period)
    {
        foreach (var asset in _db.FixedAssets.Include(a => a.Entries).ToList())
        {
            decimal accumulated = asset.Entries.Sum(e => e.Amount);
            decimal depreciable = asset.Cost - asset.SalvageValue;
            if (accumulated >= depreciable) continue;

            int year = (period.Year - asset.CommissionDate.Year) + 1;
            decimal yearly = _methods[asset.Method].Calculate(asset, year);
            decimal monthly = Math.Min(yearly / 12m, depreciable - accumulated);

            asset.Entries.Add(new DepreciationEntry
            {
                Period = period,
                Amount = monthly,
                AccumulatedDepreciation = accumulated + monthly,
                BookValue = asset.Cost - (accumulated + monthly)
            });
        }
        _db.SaveChanges();
    }
}
```

Kodda Math.Min yordamida to‘plangan eskirish amortizatsiya qilinadigan qiymatdan oshib ketishining oldi olinadi. Bu qo‘lda hisoblashda tez-tez uchraydigan xato hisoblanadi.

Buxgalteriya va soliq amortizatsiyasi o‘rtasidagi farq

Buxgalteriya hisobida amortizatsiya asosiy vositaning foydali xizmat muddati va korxona tanlagan usul asosida hisoblanadi, soliq hisobida esa qonunchilikda belgilangan tartib va me’yorlar qo‘llanadi. Shu sababli bir aktiv bo‘yicha ikki xil amortizatsiya summasi

paydo bo‘lishi mumkin. Ushbu farqni hisobga olish uchun dasturda har bir aktiv uchun ikkita parallel hisoblash saqlanishi tavsiya etiladi: moliyaviy hisobot uchun va soliq hisobi uchun. Bu yondashuv soliq deklaratsiyasini tayyorlashda vaqtinchalik farqlarni aniqlashni osonlashtiradi. Soliq me’yorlari davriy o‘zgarib turishi mumkinligi sababli, ularni kodda qat’iy yozish o‘rniga ma’lumotlar bazasidagi sozlamalar jadvalida saqlash maqsadga muvofiq.

Xavfsizlik, audit va dasturni sinash

Buxgalteriya ma’lumotlari maxfiy va muhim bo‘lgani uchun tizimda rolga asoslangan kirish nazorati qo‘llaniladi: buxgalter hisoblash va yozuvlarni kiritadi, rahbar hisobotlarni ko‘radi, administrator esa foydalanuvchilarni boshqaradi. Har bir o‘zgartirish (aktiv qo‘shish, usulni almashtirish, davrni yopish) kim tomonidan va qachon bajarilgani bilan audit jurnaliga yoziladi. Yopilgan davr ma’lumotlarini o‘zgartirish bloklanadi, zarurat bo‘lsa alohida tuzatish yozuvi kiritiladi. Parollar xeshlangan holda saqlanadi.

Hisoblash mantiqining to‘g‘riligini tekshirish uchun birlik testlaridan (xUnit) foydalanish mumkin. Masalan, yuqoridagi shartli misol uchun to‘g‘ri chiziqli usul birinchi yilda 21 600 000 so‘m, kamayuvchi qoldiq usuli esa 48 000 000 so‘m bersa, beshinchi yil oxirida to‘plangan eskirish ikkala holatda ham 108 000 000 so‘mga teng bo‘lishi shart. Bunday testlar usullarni o‘zgartirganda yoki yangi usul qo‘shganda xatolarni darhol aniqlaydi.

Tizimning cheklovlari va rivojlantirish yo‘nalishlari

Taklif etilgan model hozircha asosiy vositalar hisobi bilan cheklangan va quyidagi yo‘nalishlarda kengaytirilishi mumkin:

1. aktivlarni qayta baholash va qiymatning pasayishini (impairment) hisobga olish;
2. aktivni ta’mirlash va modernizatsiya qilishda boshlang‘ich qiymatni o‘zgartirish;
3. boshqa modullar (ish haqi, ombor, schyotlar rejasi) bilan yagona ma’lumotlar bazasida integratsiya;
4. hisobotlarni PDF va Excel formatlarida eksport qilish;
5. 1C kabi mavjud dasturlardan ma’lumotlarni import qilish.

Integratsiya amalga oshirilgach, tizim moliyaviy hisobotlarni (balans, moliyaviy natijalar to‘g‘risidagi hisobot) to‘liq avtomatik shakllantirish imkonini beradi.

Xulosa

Tadqiqot natijasida asosiy vositalar amortizatsiyasi hisobini 1C va SQL Server asosida avtomatlashtirish modeli ishlab chiqildi. Strategiya namunasidan foydalanish yangi hisoblash usullarini oson qo‘shishga, buxgalteriya yozuvlarini avtomatik shakllantirish esa hisob jarayonini tezlashtirib, xatolarni kamaytirishga imkon beradi. Kelgusida modelni soliq amortizatsiyasi me’yorlari, aktivlarni qayta baholash va moliyaviy hisobotlarni (balans, moliyaviy natijalar) shakllantirish modullari bilan kengaytirish maqsadga muvofiq.

Foydalanilgan adabiyotlar

1. O‘zbekiston Respublikasining «Buxgalteriya hisobi to‘g‘risida»gi Qonuni. Lex.uz milliy qonunchilik bazasi.
2. 5-son «Asosiy vositalar» buxgalteriya hisobi milliy standarti (BHMS). Lex.uz.
3. O‘zbekiston Respublikasi Soliq kodeksi. Lex.uz.
4. Microsoft. Entity Framework Core documentation. learn.microsoft.com.
5. Microsoft. C# documentation. learn.microsoft.com.
6. Gamma E., Helm R., Johnson R., Vlissides J. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, 1994.
7. Skeet J. C# in Depth. Manning Publications.