

DASTURLASHDA ALGEBRAIK TUSHUNCHALARNING QO‘LLANILISHI

*Ajiniyoz nomidagi NDPI Fizika va
Matematika fakulteti talabasi
Dinora Sobirova*

Annotatsiya: Ushbu maqolada dasturlash jarayonida algebraik tushunchalarning — ayniqsa, ko‘paytmalar, tenglamalar, funksiyalar va guruhlar nazariyasining — amaliy qo‘llanilishi yoritiladi. Algebraik yondashuvlar algoritmlar tuzishda, dasturiy strukturalarni modellashtirishda va muammolarni optimallashtirishda muhim rol o‘ynaydi. Shuningdek, matematik mantiq va abstraktsiya asosida samarali va mustahkam kod yozish imkoniyatlari tahlil qilinadi.

Kalit so‘zlar: algebra, dasturlash, funksiyalar, algoritmlar, tenglama, struktura, guruh nazariyasi, optimallashtirish, matematik mantiq, kod abstraktsiyasi.

Zamonaviy dasturlash nafaqat amaliy ko‘nikmalarni, balki chuqur matematik bilimlarni ham talab qiladi. Dasturchi algoritmlar tuzayotganda yoki murakkab dasturiy yechimlar ustida ishlayotganda, matematik mantiq va abstrakt tushunchalarga tayanadi. Ayniqsa, algebra fani dasturlashning nazariy asosini tashkil etuvchi muhim yo‘nalishlardan biridir. Algebraik tushunchalar, jumladan, o‘zgaruvchilar bilan ishlash, tenglamalarni yechish, funksiyalarni aniqlash, ko‘paytmalar va formulalarni tahlil qilish, algoritmlarni yaratishda va dasturiy modellarni shakllantirishda keng qo‘llaniladi. Dasturlashda algebra nafaqat sonlar ustida amallar bajarish uchun, balki ma’lumotlar strukturasini yaratish, ularni samarali boshqarish, shuningdek, murakkab muammolarni soddalashtirish uchun ham zarurdir. Masalan, guruhlar nazariyasi va boshqa abstrakt algebraik strukturalar kriptografiya, grafik algoritmlari, kompyuter algebrasi va sun’iy intellekt kabi ilg‘or yo‘nalishlarda keng qo‘llaniladi. Funksiyalar va ularning xossalriga asoslangan dasturlash paradigmatlari esa zamonaviy dasturlash tillarining (masalan, funksional dasturlashda) asosiy tamoyillaridan biridir. Shuningdek, algebraik fikrlash dasturchiga kodni modul, moslashuvchan va takror foydalanish mumkin bo‘lgan tarzda yozishga yordam beradi. Bu esa dasturiy ta’minot sifatini oshirishda, xatoliklarni kamaytirishda va texnik xizmatni yengillashtirishda muhim ahamiyatga ega. Dasturlashda matematik abstraktsiyalar bilan ishlash imkoniyati tufayli murakkab real muammolarni oddiy matematik modelga keltirib, samarali algoritmlar yordamida hal qilish mumkin bo‘ladi. Mazkur maqolada algebraik tushunchalarning dasturlashdagi o‘rni, ularning konkret misollar orqali qo‘llanilishi hamda nazariy va amaliy jihatdan qanday afzalliklar berishi atroflicha yoritiladi.

1. Algebraik tushunchalarning dasturlashdagi o‘rni

Dasturlashda algebraik tushunchalar turli darajadagi masalalarni hal qilishda markaziy rol o'ynaydi. O'zgaruvchilar, konstantalar, tenglamalar va funksiyalar dasturlarning asosiy elementlari sifatida xizmat qiladi. Misol uchun, C++, Java, Python kabi tillarda dastur yozayotganda qiymatlarni o'zgaruvchilarga tayinlash (assignment), shartli ifodalar (conditional expressions) yoki sikllarda (loops) algebraik munosabatlar doimiy ravishda ishlatiladi. Dasturlash tillarida ishlatiladigan sintaksis va semantika ham algebraik qoidalarga asoslanadi. Masalan, $x = a + b * c$ ko'rinishidagi ifoda faqat dasturlash amali bo'lib qolmay, ayni paytda algebraik ifodaning ham amaliy ko'rinishidir. Bu kabi munosabatlar orqali murakkab dasturiy strukturani soddalashtirish, modullashtirish va qayta ishlatish imkoniyati vujudga keladi.

2. Funksiyalar va ularning modullashtirishdagi roli

Algebraik funksiyalar dasturlashda alohida modul sifatida qo'llaniladi. Har bir funksional modul mustaqil ravishda ishlaydi va bu tizimni tushunish, sinovdan o'tkazish va xizmat ko'rsatishni osonlashtiradi. Masalan, $f(x) = x^2 + 2x + 1$ kabi algebraik ifoda Python dasturlash tilida quyidagicha yozilishi mumkin:

```
python
def f(x):
    return x**2 + 2*x + 1
```

Bu ko'rinishda funksiyalar yordamida kodni qismlarga ajratish va ularni boshqa joylarda qayta ishlatish imkoniyati mavjud bo'ladi. Bunday modullashtirish dasturiy ta'minotni tizimli loyihalashda muhim o'rin egallaydi.

3. Tenglamalar va algoritmlarni optimallashtirish

Ko'p hollarda dasturlash muammosi biror algebraik tenglamani yechishga olib keladi. Masalan, chiziqli yoki kvadratik tenglamalarni yechish algoritmlari ko'plab dasturiy tizimlarda, jumladan muhandislik, iqtisodiyot va fizika sohalarida keng qo'llaniladi. Quyidagi Python kodi kvadratik tenglama yechimini ifodalaydi:

```
python
import math

def kvadrat_tenglama(a, b, c):
    d = b**2 - 4*a*c
    if d > 0:
        x1 = (-b + math.sqrt(d)) / (2*a)
        x2 = (-b - math.sqrt(d)) / (2*a)
        return x1, x2
    elif d == 0:
        x = -b / (2*a)
        return x
    else:
        return "Real yechimlar mavjud emas"

# Misol: 1x^2 + 2x + 1 = 0
print(kvadrat_tenglama(1, 2, 1))
```

Bu yerda diskriminant formulasi algebraik yechimdan foydalangan holda dasturlashtirilgan.

4. Guruhlar nazariyasi va kriptografiya qo'llanishi

Abstrakt algebra, xususan guruhlar nazariyasi, zamonaviy kriptografik algoritmlarda katta rol o'ynaydi. RSA algoritmi, Diffie-Hellman almashinuvi, elliptik egri chiziqlar asosidagi kriptografiya — bularning barchasi algebraik strukturalarga tayanadi. Ular orqali ma'lumotlarni shifrlash, xavfsiz uzatish va autentifikatsiyani ta'minlash mumkin. Misol uchun, modulyar arifmetika (ya'ni qoldiq bo'yicha hisoblash) orqali shifrlash algoritmlari tuziladi. Bu yerda $a \equiv b \pmod{m}$ kabi algebraik ifodalar bevosita dasturiy kodda ifodalanadi va kriptografik himoya tizimlarining asosini tashkil etadi.

5. Algebraik abstraktsiya va kod sifatini oshirish

Dasturchi algebraik abstraktsiyalarni tushunganda, kod tuzilmasi mukammallashadi, sintaktik jihatdan soddalashadi va semantik jihatdan aniqlikka ega bo'ladi. Algebraik yondashuv kodni takrorlanmas (DRY – Don't Repeat Yourself) va tahlil qilishga yaroqli qiladi. Bu esa katta dasturiy loyihalarda samaradorlikni oshiradi.

Dasturlash — bu faqat texnik ko'nikmalar to'plami emas, balki chuqur nazariy asosga ega bo'lgan murakkab jarayondir. Ushbu maqolada ko'rib chiqilganidek, algebra fani aynan shu nazariy asosni ta'minlovchi muhim fanlardan biridir. Algebraik tushunchalar, ya'ni funksiyalar, tenglamalar, o'zgaruvchilar, guruhlar nazariyasi va modulyar arifmetika dasturlashning turli bosqichlarida — algoritm yaratish, strukturaviy kod yozish, kriptografik himoya tizimlarini qurish hamda optimallashtirish jarayonlarida keng qo'llaniladi. Amaliy jihatdan qaralganda, algebraik yondashuvlar dasturchiga real muammolarni matematik modellash va samarali algoritmlarni ishlab chiqish imkonini beradi. Funksiyalar orqali kod modullashtiriladi, tenglamalar asosida avtomatlashtirilgan hisob-kitoblar amalga oshiriladi, guruhlar nazariyasi yordamida esa murakkab kriptografik muammolar hal etiladi. Algebraik fikrlash ko'nikmalari dasturiy ta'minot sifatini oshirish, kodni o'qilishi oson va takror foydalanishga yaroqli qilishda muhim rol o'ynaydi. Shuningdek, ushbu yondashuv nafaqat nazariy, balki amaliy dasturiy ta'lim jarayonida ham qo'llanilishi zarur. Shuning uchun, dasturlashni o'rganayotgan talabalar va mutaxassislar algebra faniga chuqur e'tibor qaratishlari lozim. Algebraik bilimlar nafaqat muammolarni tushunishda, balki ularni kreativ tarzda hal qilishda ham keng imkoniyatlar ochadi. Xulosa qilib aytganda, algebraik tushunchalarning dasturlashdagi qo'llanilishi zamonaviy axborot texnologiyalari rivojlanishida fundamental asos bo'lib xizmat qiladi. Shuning uchun, ularni chuqur o'zlashtirish va amaliyotda qo'llash zamon talabi hisoblanadi.

Foydalanilgan adabiyotlar:

1. Karimov, S., & Abdullayev, H. (2019). *Algoritmlar va dasturlash asoslari*. Toshkent: “Fan va texnologiya” nashriyoti.
2. G‘ulomov, S. (2020). *Abstrakt algebra asoslari*. Toshkent: O‘zbekiston milliy universiteti nashriyoti.
3. Kattamov, I. & Raxmonov, J. (2021). *Matematik mantiq va algoritmik tafakkur asoslari*. Urganch: UrDU nashriyoti.

