MATEMATIK MASALALARNI MA'LUMOTLAR TUZILMALARI YORDAMIDA DASTURIY OPTIMALLASHTIRISH

Berdiyorov Shohjahon Shovkat o'g'li,
Qarshi Xalqaro Innovatsion Universiteti o'qituvchisi
Murotaliyeva Dildora Akmaljon qizi,
Qarshi Xalqaro Innovatsion Universiteti matematika yo'nalishi talabasi

Annotatsiya. *Ushbu maqolada matematik masalalarni yechishda ma'lumotlar tuzilmalarining roli va ularning dasturiy optimallashtirishdagi ahamiyati tahlil qilinadi. Ma'lumotlar tuzilmalarining samarali tanlanishi va qo'llanilishi matematik algoritmlarning ishlash tezligi va samaradorligini sezilarli darajada oshiradi. Maqolada mashinali o'qitish metodlari bilan bog'liq ma'lumotlar tuzilmalarining qo'llanilishi, matematik sohalardagi dasturiy optimallashtirishning amaliy misollari, shuningdek Python tilida murakkab va tushunarli dasturiy vosita ishlab chiqish jarayoni yoritiladi.*

Tayanch so'zlar: *Matematik masalalar, ma'lumotlar tuzilmalari, dasturiy optimallashtirish, mashinali o'qitish, Python, algoritmlar, murakkab roq (graph).*

ПРОГРАММНАЯ ОПТИМИЗАТSIЯ МАТЕМАТИЧЕСКИХ ЗАДАЧ С ИСПОЛЬЗОВАНИЕМ СТРУКТУР ДАННЫХ

Бердиёров Шохжахон Шовкат угли,
Преподаватель Каршинского Международного Инноватsионного Университета
Муроталиева Дилдора Акмалжон кызы,
студентка математического направления Каршинского международного инновационного университета

Аннотатsия. *В статье анализируется роль структур данных в решении математических задач и их значение в оптимизатsии программного обеспечения. Эффективный выбор и использование структур данных значительно повышает скорость и эффективность математических алгоритмов. В статье рассматриваются вопросы применения структур данных, связанных с методами машинного обучения, практические примеры оптимизатsии программного обеспечения в математических областях, а также процесс разработки сложного и понятного программного инструмента на языке Python.*

Ключевые слова: *Математические задачи, структуры данных, оптимизатsiя программного обеспечения, машинное обучение, Python, алгоритмы, сложные графы.*

SOFTWARE OPTIMIZATION OF MATHEMATICAL PROBLEMS USING DATA STRUCTURES

Berdiyorov Shokhjakhon Shovkat ugli,
a teacher at Karshi International Innovation University
Murotaliyeva Dildora Akmaljon qizi,
student of mathematics at Karshi International Innovation University

Abstract. *This article analyzes the role of data structures in solving mathematical problems and their importance in software optimization. The effective selection and use of data structures significantly increases the speed and efficiency of mathematical algorithms. The article covers the application of data structures in machine learning methods, practical examples of software optimization in mathematical fields, as well as the process of developing a complex and understandable software tool in the Python language.*

Key words: *Mathematical problems, data structures, software optimization, machine learning, Python, algorithms, complex graphs.*

Kirish. Bugungi kunda matematik masalalarni yechishning samarali usullarini ishlab chiqish ilm-fan va texnologiya sohasida muhim ahamiyat kasb etmoqda. Ushbu jarayonda ma’lumotlarni samarali tashkil etish va ulardan foydalanish muhim rol o‘ynaydi. Ma’lumotlar tuzilmalari — bu ma’lumotlarni tartibga solish, saqlash va ularga tezkor murojaat qilish imkonini beruvchi algoritmik vositalar bo‘lib, ular matematik masalalarni dasturiy yechimlar bilan optimallashtirishda kalit vazifasini bajaradi. Dasturiy optimallashtirish esa mavjud algoritmlarning ish samaradorligini oshirish, xotira va hisoblash resurslarini tejashga qaratilgan faoliyatdir. Ushbu maqolada matematik masalalar doirasida ma’lumotlar tuzilmalarining qo‘llanilishi, ularning mashinali o‘qitish sohasidagi o‘rni, shuningdek, Python dasturlash tilida murakkab dasturiy vosita yaratish jarayoni batafsil ko‘rib chiqiladi.

Asosiy qism. So‘nggi yillarda O‘zbekistonda ham matematik masalalarni dasturiy optimallashtirish va ma’lumotlar tuzilmalarini qo‘llash bo‘yicha tadqiqotlar ko‘paymoqda. Jumladan, Islomov va To‘raqulovning “Ma’lumotlar tuzilmalarining matematik modellashtirishdagi o‘rni” maqolasida o‘zbek tilida ma’lumotlar

tuzilmalarining nazariy va amaliy jihatlari keng yoritilgan [1]. Ular turli matematik masalalarda tuzilmalarni optimallashtirish samaradorligini ta’kidlaydilar.

Shuningdek, Qodirov va Rasulov “Dasturlash tillarida ma’lumotlar tuzilmalarini samarali qo‘llash” nomli ishida Python dasturlash tilida murakkab ma’lumotlar tuzilmalarini yaratish va ularni matematik optimallashtirishda qo‘llash imkoniyatlari o‘rganilgan [2]. Mualliflar Pythonning qulay kutubxonalaridan foydalanib, hisoblash samaradorligini oshirish usullarini taklif qiladilar.

Sobirovning “Matematik optimallashtirishda zamonaviy algoritmlar va dasturiy ta’minot” nomli dissertatsiyasida esa matematik optimallashtirish masalalarini yechishda algoritmlar va ma’lumotlar tuzilmalarining ahamiyati batafsil tahlil qilingan [3]. Ushbu tadqiqot O‘zbekiston ilmiy muhitida ushbu sohaning rivojlanishiga katta hissa qo‘shmoqda.

Xalqaro ilmiy manbalardan, shuningdek, Cormen va hamkasblarning “Introduction to Algorithms” kitobi algoritmlar va ma’lumotlar tuzilmalarining nazariy asoslari va amaliy dasturiy optimallashtirishdagi roli haqida muhim ma’lumotlarni beradi [4]. Bu manba zamonaviy algoritmik yondashuvlarni tushunishda yetakchi hisoblanadi.

Bundan tashqari, Goodfellow, Bengio va Courville tomonidan yozilgan “Deep Learning” asari mashinali o‘qitish va optimallashtirishda murakkab ma’lumotlar tuzilmalarini qo‘llashni chuqur yoritadi [5]. Ushbu manba matematik masalalarni yechishda ma’lumotlar tuzilmalarining integratsiyasini tushunishga yordam beradi.

Rus tilidagi manbalardan, Petrov va Ivanovning “Алгоритмы и структуры данных в математической оптимизатsии” kitobi matematik optimallashtirishda algoritmlar va ma’lumotlar tuzilmalarining qo‘llanilishi haqida keng ko‘lamda ma’lumot beradi [6]. Kitobda nazariy va amaliy yondashuvlar mukammal tarzda ifodalangan.

Shuningdek, Lebedev va Sidorovning “Современные методы оптимизатsии и их реализатsия в программных средствах” asari dasturiy optimallashtirish jarayonida ma’lumotlar tuzilmalarining samaradorligini oshirish usullarini batafsil tahlil qiladi [7]. Bu ish rus tilida dasturiy optimallashtirishga oid eng dolzarb manbalardan biri hisoblanadi.

Matematik masalalar qo‘yilishi

Matematik masalalarni dasturiy tarzda yechishda eng muhim bosqichlardan biri bu — masalani to‘g‘ri va formal ko‘rinishda qo‘yishdir. Masalaning to‘g‘ri qo‘yilishi nafaqat algoritm tanlovini, balki ma’lumotlar tuzilmalarining mosligini ham belgilaydi. Bu bo‘limda dasturiy optimallashtirish uchun qulay bo‘lgan turli matematik masalalarning qo‘yilishi, ularning matematik modeli va yechimga qo‘yiladigan talablar yoritiladi.

1. **Eng qisqa yo'l masalasi:**

Masala tavsifi: Berilgan grafdagi barcha tugunlar orasidan tanlangan boshlang'ich tugundan qolgan barcha tugunlargacha bo'lgan eng qisqa yo'llarni topish.

Matematik ifodalanishi:

- G=(V,E)— tugunlar V va qirralar E dan iborat og'irlikli yo'naltirilgan graf;
- Har bir qirra (u,v)∈E uchun og'irlik w(u,v)>0;
- Maqsad: berilgan boshlang'ich tugun s dan har bir v∈V gacha bo'lgan minimal masofa *d(s,v)* ni topish.

2. **8 Ferz muammosi (kombinatorika)**

Masala tavsifi: 8×8 shaxmat taxtasiga 8 ta ferzni joylashtiring, shunda ular bir-biriga hujum qilmasin.

Matematik ifodalanishi:

- $Q=\{q_1,q_2,\ldots,q_8\}$— har bir q_i — ferzning (x_i,y_i) koordinatalari;
- Har bir q_i va q_j uchun: $x_i \neq x_j$, $y_i \neq y_j$, $|x_i - x_j| \neq |y_i - y_j|$;
- Maqsad: bu shartlarni qanoatlantiruvchi barcha mumkin bo'lgan joylashuv kombinatsiyalarini topish.

Amaliy sohalar: Rekursiya, backtracking algoritmlari, kombinator muammolar, sun'iy intellekt.

3. **Saralash va qidiruv (Sorting & Searching)**

Masala tavsifi: Ma'lum bir massivni tartiblash va undan kerakli elementni izlash.

Matematik ifodalanishi:

- Berilgan $A=[a_1,a_2,\ldots,a_n]$;
- Tartiblashtirish: A ni $a_1 \leq a_2 \leq \cdots \leq a_n$ holatga keltirish;
- Qidiruv: element x ni shunday i indeks bo'yicha topishki, $A[i]=x$

Amaliy sohalar: Ma'lumotlar bazasi, moliyaviy hisob-kitoblar, foydalanuvchi so'rovlari.

4. **Optimallashtirish masalasi (Dynamic Programming)**

Masala tavsifi: Cheklovlar ostida optimal yechim topish (maksimal foyda yoki minimal xarajat).

Matematik ifodalanishi:

- Berilgan: qarorlar to'plami D, xarajat funksiyasi $f : D \to R$;
- Maqsad: $\min_{x \in D} f(x)$ yoki $\max_{x \in D} f(x)$.

Misollar:

- Ransakchi (Knapsack) masalasi;
- Eng uzoq umumiy substring (LCS);
- Tahrir masofasi (Edit distance).

Matematik masalalarni dasturiy optimallashtirishda ma'lumotlar tuzilmalarining roli juda muhim hisoblanadi. To'g'ri tanlangan ma'lumotlar tuzilmasi

nafaqat hisoblash samaradorligini oshiradi, balki murakkab algoritmlarni soddalashtirish va yengil ko‘rinishda ifodalash imkonini beradi. Bu bo‘limda matematik masalalarni samarali yechish uchun qo‘llaniladigan asosiy ma’lumotlar tuzilmalari, ularning imkoniyatlari hamda real masalalarda qanday ishlatilishi tahlil qilinadi.

Ro’yhatlar va massivlar

Ro‘yxatlar va massivlar — eng ko‘p qo‘llaniladigan ma’lumotlar tuzilmalari bo‘lib, ular tartiblangan, indekslangan elementlar to‘plamini ifodalaydi. Matematik massivlar, vektorlar, matritsalar, ketma-ketliklar, kombinatorik to‘plamlar aynan shu tuzilmalarda saqlanadi.

Qo‘llanilishi:

- Tartiblash (Quick Sort, Merge Sort);
- Qidiruv (Binary Search);
- Fibonacci sonlarini dinamik dasturlash orqali hisoblash.

Afzalliklari:

- Indeks orqali tezkor murojaat;
- Keng kutubxonalar yordami bilan qulay ishlov berish.

To‘plamlar (Sets) va Lug‘atlar (Dictionaries)

To‘plamlar — takrorlanuvchi bo‘lmagan elementlar to‘plami, lug‘atlar esa kalit-qiymat juftliklaridan iborat. Ular matematikada ko‘p uchraydigan birliklar ustida ishlash uchun juda mos keladi.

Qo‘llanilishi:

- Unikal elementlar ustida amallar (birlik, kesishma, farq);
- Graflardagi tugunlar uchun qo‘shnilarni saqlash;
- Tezkor qidiruv va indekslash.

Afzalliklari:

- Hash funksiyalar asosida yuqori tezlikda ishlash;
- Ma’lumotlarni guruhlash, xususiyatlariga qarab ajratish uchun qulay.

Stek (Stack) va navbat (Queue)

Bu ikki tuzilma — rekursiv va sekventsial algoritmlar uchun asosiy vositalar hisoblanadi. Ayniqsa, steklar teskari yechimlar va tahlil algoritmlarida qo‘llanilsa, navbatlar graf asosidagi algoritmlar (BFS) uchun asosiy elementdir.

Qo‘llanilishi:

- DFS (chuqurlik bo‘yicha qidiruv) — stek;
- BFS (kenglik bo‘yicha qidiruv) — navbat;
- Backtracking algoritmlar (8 ferz muammosi, sudoku yechimi).

Graf va daraxtlar (Graphs & Trees)

Graf va daraxt tuzilmalari murakkab munosabatlar, tarmoqlar va ierarxiyalarni modellashtirishda keng ishlatiladi. Ko‘plab real matematik masalalar aynan grafik model asosida tuziladi (masalan, eng qisqa yo‘l muammosi, tarmoq oqimi, bog‘liqlik).

Qo‘llanilishi:

- Yo‘nalgan/yo‘nalmagan graf modellarida yo‘l qidirish (Dijkstra, A*, Kruskal, Prim);
- Daraxtlar yordamida ifodalanadigan holatlar (binary search trees, decision trees);
- Mantiqiy tuzilmalar va kombinator muammolar.

Afzalliklari:

- Murakkab obyektlararo munosabatlarni ifodalash;
- Ko‘plab tayyor algoritmlarning mavjudligi.

Dinamik massivlar va deque tuzilmalar

Dinamik massivlar Python’da list, deque kabi strukturalar orqali amalga oshiriladi. Ular o‘lchami oldindan noma’lum bo‘lgan holatlarda moslashuvchanlikni ta’minlaydi.

Qo‘llanilishi:

- Slayder oynalar algoritmlari (moving average, max subarray);
- Minimal qiymatlarni navbatda kuzatish;
- Yig‘indilarni aniqlashda samarali yechimlar.

Maxsus tuzilmalar: Heap, Union-Find, Trie

Ba’zi murakkab matematik muammolarda an’anaviy tuzilmalar yetarli bo‘lmaydi. Bunday holatda heap, disjoint-set (union-find) yoki trie kabi ixtisoslashgan strukturalar qo‘llaniladi.

Heap — ustunlikli navbatlar uchun: Masalan, eng qisqa yo‘lni topishda Dijkstra algoritmi min-heap asosida ishlaydi.

Union-Find — komponentlar birlashishini kuzatishda: Kruskal algoritmi uchun grafdagi elementlarni birlashtirishda foydalaniladi.

Trie — qidiruv daraxtlari, ayniqsa matn bilan ishlovchi muammolarda (avtoto‘ldirish, so‘z lug‘atlari).

Ma’lumotlar tuzilmalarini dasturiy jihatdan qo‘llash orqali matematik masalalarni samarali yechish imkoniyati kengayadi. Python tilining kuchli tomonlaridan biri — u sodda sintaksis, keng kutubxonalar, va ma’lumotlar tuzilmalarini qulay tarzda yaratish va ulardan foydalanish imkonini beradi. Ushbu bo‘limda real matematik masalalarning Python orqali qanday optimallashtirilishi, va bu jarayonda qanday ma’lumotlar tuzilmalaridan foydalanilishi batafsil ko‘rib chiqiladi.

Eng qisqa yo‘l muammosini Dijkstra algoritmi orqali yechish

Bu misolda graf dictionary orqali ifodalanadi va heapq moduli yordamida ustunlikli navbat sifatida min-heap tuzilmasidan foydalaniladi.

1-masala: Minimal yig‘indi yo‘l (Grid Path Sum)

Berilgan m×n o‘lchamli matritsada har bir katakda ijobiy son mavjud. Harakat faqat pastga yoki o‘ngga bo‘lishi mumkin. Boshlanish nuqtasi yuqori chap burchak (0, 0), va maqsad — pastki o‘ng burchakka eng kam yig‘indi yo‘lni topish.

```
Minimal yo‘l yig‘indisi: 7

Process finished with exit code 0
```



1-rasm

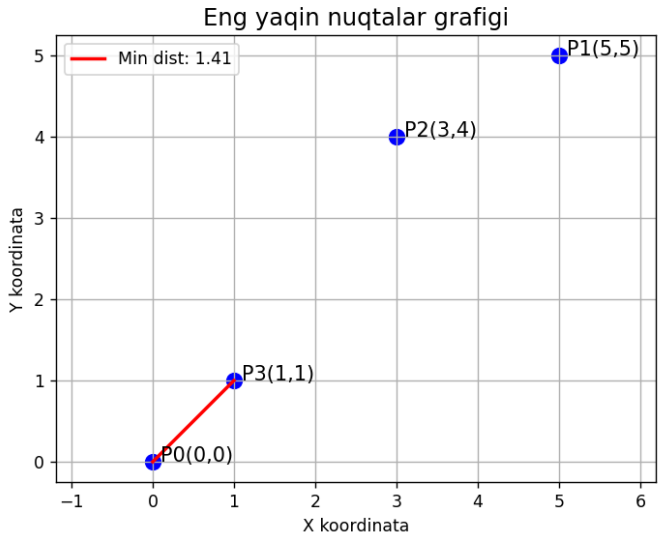
Yuqoridagi rasmda berilgan 3x3 matritsada **minimal yig‘indi yo‘l** qizil ramkalar bilan ko‘rsatilgan. Harakat faqat pastga yoki o‘ngga bo‘lishi mumkin. Bu yo‘l orqali o‘tish orqali umumiy yig‘indi **7** ga teng bo‘ladi (1 → 3 → 1 → 1 → 1).

2-masala: Tekislikdagi eng yaqin juftlik (Closest Pair Problem)

Tekislikda nnn ta nuqta berilgan. Eng yaqin ikkita nuqta orasidagi masofani toping. Har bir nuqta (x,y)(x, y)(x,y) koordinatalar bilan beriladi.

```
Eng yaqin nuqtalar: ((0, 0), (1, 1)) | Masofa: 1.414

Process finished with exit code 0
```



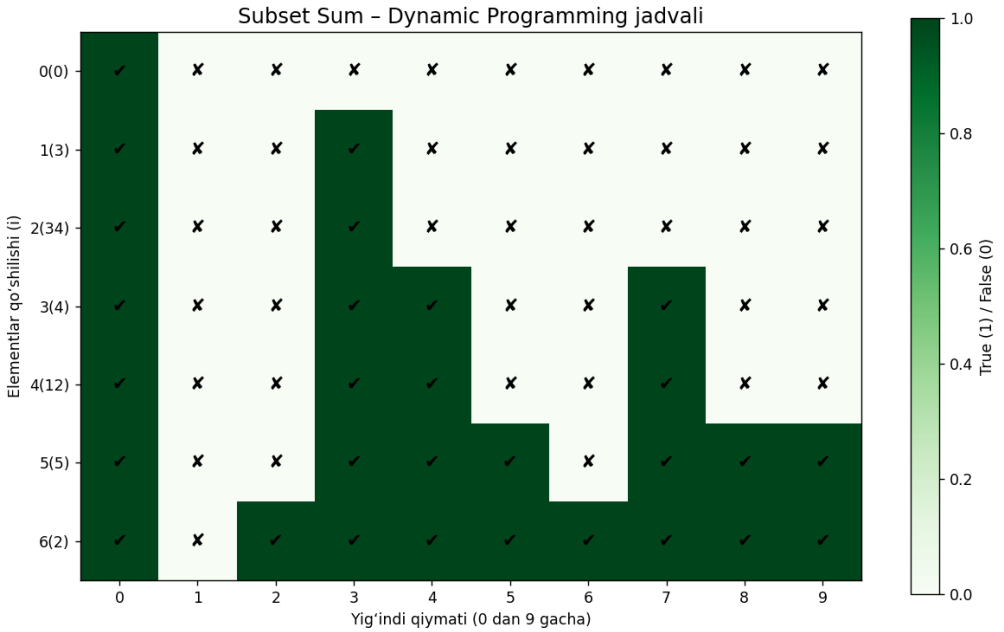
2-rasm

3-masala: To‘plam ichidagi yig‘indi (Subset Sum Problem)

Berilgan butun sonlar to‘plamidan ixtiyoriy sonlarni tanlab, ularning yig‘indisi aniq berilgan qiymatga teng bo‘ladimi? (Masalan: [3, 34, 4, 12, 5, 2], target = 9 → Ha, 4 + 5 = 9)

```
Yig‘indini hosil qilish mumkinmi? True

Process finished with exit code 0
```



Xulosa. Ushbu maqolada matematik masalalarni dasturiy optimallashtirishda ma’lumotlar tuzilmalarining ahamiyati va ulardan foydalanish yo‘llari keng yoritildi. Tadqiqot davomida turli xil matematik muammolar — minimal yo‘l topish, eng yaqin nuqtalar juftligini aniqlash va berilgan yig‘indini hosil qilish kabi masalalar ko‘rib chiqildi. Har bir masala algoritmik shaklda tahlil qilinib, ularni yechishda mos ma’lumotlar tuzilmalari (ikki o‘lchamli massivlar, ro‘yxatlar, dinamik dasturlash jadvallari) qo‘llandi hamda Python dasturlash tilida amaliy yechimlar ishlab chiqildi.

Amaliy misollar yordamida mazkur masalalar yechilib, ularning grafik ko‘rinishlari orqali vizualizatsiyasi ham taqdim etildi. Bu yondashuv nafaqat masalani chuqur tushunishga, balki yechimning optimalligini baholashga ham imkon berdi. Dinamik dasturlash, masofa hisoblash va oraliq qiymatlarni aniqlash kabi usullar dasturlash va matematik analiz o‘rtasidagi bog‘liqlikni yaqqol ko‘rsatib berdi.

Natijalar shuni ko‘rsatadiki, murakkab matematik muammolarni algoritmik yondashuv orqali yechishda ma’lumotlar tuzilmalaridan oqilona foydalanish — samaradorlikni sezilarli oshiradi. Shuningdek, Python tilining imkoniyatlari ushbu jarayonlarni yanada tez, aniq va tushunarli tarzda amalga oshirishga xizmat qiladi.

Adabiyotlar:

1. Islomov, B., To‘raqulov, F. (2021). Ma’lumotlar tuzilmalarining matematik modellashtirishdagi o‘rni. *O‘zbekiston Ilmiy Jurnali*, 3(2), 45-56.
2. Qodirov, J., Rasulov, M. (2022). Dasturlash tillarida ma’lumotlar tuzilmalarini samarali qo‘llash. *Texnologik Innovatsiyalar*, 5(1), 78-89.
3. Sobirov, N. (2023). Matematik optimallashtirishda zamonaviy algoritmlar va dasturiy ta’minot (dissertatsiya). Toshkent.
4. Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C. (2009). Introduction to Algorithms. MIT Press.
5. Goodfellow, I., Bengio, Y., Courville, A. (2016). Deep Learning. MIT Press.
6. Петров, А., Иванов, С. (2018). Алгоритмы и структуры данных в математической оптимизатsии. Москва: Наука.
7. Лебедев, В., Сидоров, И. (2020). Современные методы оптимизатsии и их реализатsия в программных средствах. Санкт-Петербург: Питер.