“Pedagogik ta’lim va tarbiya transformatsiyasida fundamental, amaliy va innovatsion tadqiqotlarning konseptual asoslari” mavzusidagi Xalqaro ilmiy-amaliy anjuman materilalari to‘plami

23.06.2026

Proceedings of the International Scientific and Practical Conference on “Conceptual Foundations of Fundamental, Applied, and Innovative Research in the Transformation of Pedagogical Education and Upbringing”

LAPLAS TEOREMASI ASOSIDA DETERMINANTLARNI HISOBLASH ALGORITMLARI VA ULARNI PYTHON DASTURLASH TILIDA AMALGA OSHIRISH

Sa’dullayeva Nilufar Javlon qizi
Urganch davlat pedagogika instituti Matematika va informatika yo‘nalishi 3-kurs talabasi
nilufarsadullayeva04@gmail.com

***DOI:* https://doi.org/10.5281/zenodo.21158775**

Abstract. *This paper examines the Laplace theorem, one of the classical methods for computing determinants. Based on the Laplace expasion formula, an algorithm for computing determinants has been developed and its implementation in the Python programming language is presented. The mathematical and algorithmic aspects of determinant computation are analyzed, and the application in the field of computer science are highlighted.*

Keywords*: Determinant, Laplace theorem, minor, algebraic complement, recursion, algorithm, Python, matrix, linear algebra.*

Аннотация. *В данной статье рассматривается теорема Лапласа — один из классических методов вычисления определителей. На основе формулы разложения Лапласа разработан алгоритм вычисления определителей и представлена его реализация на языке программирования Python. Проведён анализ математических и алгоритмических аспектов вычисления определителей, а также рассмотрены области применения в информатике.*

Ключевые слова. *Определитель, теорема Лапласа, минор, алгебраическое дополнение, рекурсия, алгоритм, Python, матрица, линейная алгебра.*

Annotatsiya. *Mazkur maqolada determinantlarni hisoblashning klassik usullaridan biri bo‘lgan Laplas teoremasi o‘rganilgan. Laplas yoyish formulasi asosida determinantlarni hisoblash algoritmi ishlab chiqilgan va Python dasturlash tilida realizatsiyasi keltirilgan. Determinantlarni hisoblashning matematik va algoritmik jihatlari tahlil qilinib, informatika fanidagi qo‘llanilish sohalari yoritilgan.*

Kalit so‘zlar: *Determinant, Laplas teoremasi, minor, algebraik to‘ldiruvchi, rekursiya, algoritm, Python, matritsa, chiziqli algebra.*

Chiziqli algebra zamonaviy matematikaning eng muhim bo‘limlaridan biri bo‘lib, uning ko‘plab tushunchalari informatika, sun’iy intellekt, kompyuter grafikasi va muhandislik masalalarida qo‘llaniladi. Determinant tushunchasi kvadrat matritsalarning asosiy xarakteristikalaridan biri hisoblanadi. Determinant yordamida matritsaning teskari matritsaga ega yoki ega emasligi aniqlanadi, chiziqli tenglamalar sistemalari yechiladi hamda geometrik o‘zgartirishlar tahlil qilinadi.

Laplas teoremasida $r=1$ bo`lsa, ya’ni $D$ detirminantda bitta $i-$satr ajratilsa, u xolda $M_1, M_2, \ldots, M_t$ minorlar, birinchi tartibli minorlar sifatida bo`ladi. $A_1, A_2, .., A_t$ algebraik to`ldiruvchilar bu vaqtda $a_{i1}, a_{i2}, \ldots, a_{in}$ elementlarning $A_1, A_2, .., A_n$ algebraik to`ldiruvchilarga aylanadi. SHunday qilib, $i-$satr $a_{i1}, a_{i2}, \ldots, a_{in}$ elementlarini o`zining $A_1, A_2, .., A_n$ algebraik to`ldiruvchilarga qupaytirib (yoki $j-$ustunning

$a_{j1}, a_{j2}, \ldots, a_{jn}$ elementlarning o`zining algebraik to`ldiruvchilarga ko`paytirib) qo`shsak, xosil bo`lgan yig`indi D detirminantga teng bo`ladi. Determinantlarni hisoblash algoritmi quyidagicha bo‘ladi:
1. Matritsa o‘lchamini aniqlash.
2. 1×1 va 2×2 holatlar uchun bazaviy formulalarni qo‘llash.
3. Birinchi satr bo‘yicha yoyish.
4. Har bir element uchun minor hosil qilish.
5. Minor determinantlarini rekursiv hisoblash.
6. Natijalarni yig‘ish orqali determinant qiymatini topish.

Misol.1) $i=2$ $\quad i=3$ $\quad i=4$

$$\begin{vmatrix} 2 & -1 & 3 & 4 & -5 \\ 4 & -2 & 7 & 8 & -7 \\ -6 & 4 & -9 & -2 & 3 \\ 3 & -2 & 4 & 1 & -2 \\ -2 & 6 & 5 & 4 & -3 \end{vmatrix} = A_{234}^{123}\cdot M_{234}^{123} + A_{234}^{124}\cdot M_{234}^{124} + A_{234}^{125}\cdot M_{234}^{125} + A_{234}^{345}\cdot M_{234}^{345} + A_{234}^{245}\cdot M_{234}^{245} +$$

$$+A_{234}^{145}\cdot M_{234}^{145} + A_{234}^{234}\cdot M_{234}^{234} + A_{234}^{235}\cdot M_{234}^{235} + A_{234}^{134}\cdot M_{234}^{134} + A_{234}^{135}\cdot M_{234}^{135} = (-1^{15})\cdot\begin{vmatrix} 4 & -5 \\ 4 & -3 \end{vmatrix}\cdot\begin{vmatrix} 4 & -2 & 7 \\ -6 & 4 & -9 \\ 3 & -2 & 4 \end{vmatrix} +$$

$$+(-1^{16})\cdot\begin{vmatrix} 3 & -5 \\ 5 & -3 \end{vmatrix}\cdot\begin{vmatrix} 4 & -2 & 8 \\ -6 & 4 & -2 \\ 3 & -2 & 1 \end{vmatrix} + (-1^{17})\cdot\begin{vmatrix} 3 & 4 \\ 5 & 4 \end{vmatrix}\cdot\begin{vmatrix} 4 & -2 & -7 \\ -6 & 4 & 3 \\ 3 & -2 & -2 \end{vmatrix} + (-1^{21})\cdot\begin{vmatrix} 2 & -1 \\ -2 & 6 \end{vmatrix}\cdot\begin{vmatrix} 7 & 8 & -7 \\ -9 & -2 & 3 \\ 4 & 1 & -2 \end{vmatrix} +$$

$$+(-1^{20})\cdot\begin{vmatrix} 2 & 3 \\ -2 & 5 \end{vmatrix}\cdot\begin{vmatrix} -2 & 8 & -7 \\ 4 & -2 & 3 \\ -2 & 1 & -2 \end{vmatrix} + (-1^{19})\cdot\begin{vmatrix} -1 & 3 \\ 6 & 5 \end{vmatrix}\cdot\begin{vmatrix} 4 & 8 & -7 \\ -6 & -2 & 3 \\ 3 & 1 & -2 \end{vmatrix} + (-1^{18})\cdot\begin{vmatrix} 2 & -5 \\ -2 & -3 \end{vmatrix}\cdot\begin{vmatrix} -2 & 7 & 8 \\ 4 & -9 & -2 \\ -2 & 4 & 1 \end{vmatrix} +$$

$$+(-1^{19})\cdot\begin{vmatrix} 2 & 4 \\ -2 & 4 \end{vmatrix}\cdot\begin{vmatrix} -2 & 7 & -7 \\ 4 & -9 & 3 \\ -2 & 4 & -2 \end{vmatrix} + (-1^{17})\cdot\begin{vmatrix} -1 & -5 \\ 6 & -3 \end{vmatrix}\cdot\begin{vmatrix} 4 & 7 & 8 \\ -6 & -9 & -2 \\ 3 & 4 & 1 \end{vmatrix} + (-1^{18})\cdot\begin{vmatrix} -1 & 4 \\ 6 & 4 \end{vmatrix}\cdot\begin{vmatrix} 4 & 7 & -7 \\ -6 & -9 & 3 \\ 3 & 4 & -2 \end{vmatrix} = -84$$

$$\begin{vmatrix} 2 & -1 & 3 & 4 & -5 \\ 4 & -2 & 7 & 8 & -7 \\ -6 & 4 & -9 & -2 & 3 \\ 3 & -2 & 4 & 1 & -2 \\ -2 & 6 & 5 & 4 & -3 \end{vmatrix} \sim \begin{vmatrix} 2 & -1 & 3 & 4 & -5 \\ 0 & 0 & 1 & 0 & 3 \\ -6 & 4 & -9 & -2 & 3 \\ 3 & -2 & 4 & 1 & -2 \\ -2 & 6 & 5 & 4 & -3 \end{vmatrix} \sim \begin{vmatrix} 2 & -1 & 3 & 4 & -5 \\ 0 & 0 & 1 & 0 & 3 \\ 0 & 1 & 0 & 10 & -12 \\ 3 & -2 & 4 & 1 & -2 \\ 0 & 5 & 8 & 8 & -8 \end{vmatrix} = A_{234}^{123}\cdot M_{234}^{123} + A_{234}^{124}\cdot M_{234}^{124} +$$

$$+A_{234}^{125}\cdot M_{234}^{125} + A_{234}^{345}\cdot M_{234}^{345} + A_{234}^{245}\cdot M_{234}^{245} + A_{234}^{145}\cdot M_{234}^{145} + A_{234}^{234}\cdot M_{234}^{234} + A_{234}^{235}\cdot M_{234}^{235} + A_{234}^{134}\cdot M_{234}^{134} +$$

$$+A_{234}^{135}\cdot M_{234}^{135} = (-1^{15})\cdot\begin{vmatrix} 4 & -5 \\ 8 & -8 \end{vmatrix}\cdot\begin{vmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 3 & -2 & 4 \end{vmatrix} + (-1^{16})\cdot\begin{vmatrix} 3 & -5 \\ 8 & -8 \end{vmatrix}\cdot\begin{vmatrix} 0 & 0 & 0 \\ 0 & 1 & 10 \\ 3 & -2 & 1 \end{vmatrix} + (-1^{17})\cdot\begin{vmatrix} 3 & 4 \\ 8 & 8 \end{vmatrix}\cdot\begin{vmatrix} 0 & 0 & 3 \\ 0 & 1 & -12 \\ 3 & -2 & -2 \end{vmatrix} +$$

$$+(-1^{21})\cdot\begin{vmatrix} 2 & -1 \\ 0 & 5 \end{vmatrix}\cdot\begin{vmatrix} 1 & 0 & 3 \\ 0 & 10 & -12 \\ 4 & 1 & -2 \end{vmatrix} + (-1^{20})\cdot\begin{vmatrix} 2 & 3 \\ 0 & 8 \end{vmatrix}\cdot\begin{vmatrix} 0 & 0 & 3 \\ 1 & 10 & -12 \\ -2 & 1 & -2 \end{vmatrix} + (-1^{15})\cdot\begin{vmatrix} -1 & 3 \\ 5 & 8 \end{vmatrix}\cdot\begin{vmatrix} 0 & 0 & 3 \\ 0 & 10 & -12 \\ 3 & 1 & -2 \end{vmatrix} +$$

$$+(-1^{18})\cdot\begin{vmatrix} 2 & -5 \\ 0 & -8 \end{vmatrix}\cdot\begin{vmatrix} 0 & 1 & 0 \\ 1 & 0 & 10 \\ -2 & 4 & 1 \end{vmatrix} + (-1^{19})\cdot\begin{vmatrix} 2 & 4 \\ 0 & 8 \end{vmatrix}\cdot\begin{vmatrix} 0 & 1 & 3 \\ 1 & 0 & -12 \\ -2 & 4 & -2 \end{vmatrix} + (-1^{17})\cdot\begin{vmatrix} -1 & -5 \\ 5 & -8 \end{vmatrix}\cdot\begin{vmatrix} 0 & 1 & 0 \\ 0 & 0 & 10 \\ 3 & 4 & 1 \end{vmatrix} +$$

$$+(-1^{18})\cdot\begin{vmatrix} -1 & 4 \\ 5 & 8 \end{vmatrix}\cdot\begin{vmatrix} 0 & 1 & 3 \\ 0 & 0 & -12 \\ 3 & 4 & -2 \end{vmatrix} = -84$$

Determinant Python dasturida quyidagicha ifodaladi:

```
def determinant(A):
    n = len(A)
    if n == 1:
        return A[0][0]
    if n == 2:
```

```
        return A[0][0]*A[1][1] - A[0][1]*A[1][0]

    det = 0
    for j in range(n):
        minor = []
        for row in A[1:]:
            minor.append(row[:j] + row[j+1:])
        det += ((-1)**j) * A[0][j] * determinant(minor)
    return det
```

Determinantlarni hisoblash algoritmlari kompyuter grafikasi, mashinali o‘qitish, ma’lumotlarni qayta ishlash, kriptografiya va sonli usullarda keng qo‘llaniladi. Rekursiv algoritmlar esa dasturlash fanining muhim tushunchalaridan biri hisoblanadi.

Laplas usuli kichik o‘lchamli matritsalar uchun qulay va tushunarli hisoblanadi. Biroq matritsa o‘lchami ortishi bilan hisoblashlar soni juda tez ko‘payadi. Shu sababli katta o‘lchamli matritsalar uchun Gauss usuli samaraliroq hisoblanadi.

Xulosa qilib aytsak Laplas teoremasi determinantlarni hisoblashning muhim nazariy asoslaridan biridir. Uning algoritmik talqini matematika va informatika integratsiyasini namoyon etadi. Python dasturlash tilida ushbu algoritmni amalga oshirish talabalarning ham matematik, ham dasturlash ko‘nikmalarini rivojlantirishga xizmat qiladi.

Foydalanilgan adabiyotlar

1. Sh.A.Ayupov, B.A.Omirov A.X.Xudoyberdiyev. Chiziqli Algebra. “Muxr-Press” Toshlent,2023.-392 b
2. David C. Lay. *Linear Algebra and Its Applications*. Boston: Addison-Wesley, 2012. – 576 p.
3. Howard Anton. *Elementary Linear Algebra*. New York: John Wiley & Sons, 2010. – 768 p.
4. Al Sweigart. *Automate the Boring Stuff with Python: Practical Programming for Total Beginners*. San Francisco: No Starch Press, 2019. – 592 p.
5. Joel Grus. *Data Science from Scratch: First Principles with Python*. Sebastopol, CA: O'Reilly Media, 2019. – 406 p.
6. Sh. Xurramov. *Chiziqli algebra va analitik geometriya*. Toshkent: O‘zbekiston Respublikasi Oliy va o‘rta maxsus ta’lim vazirligi, 2008. – 320 b.
7. Bernard Kolman, David R. Hill. *Elementary Linear Algebra*. Upper Saddle River, NJ: Prentice Hall, 2000. – 608 p.