Київський національний університет імені Тараса Шевченка

Факультет комп`ютерних наук та кібернетики

Кафедра інтелектуальних інформаційних систем

Алгоритми та складність

Завдання №5

“Алгоритм Штрассена”

Виконав студент 2-го курсу

Групи К-28

Самойлич Євгеній Костянтинович

2020

**Завдання**:

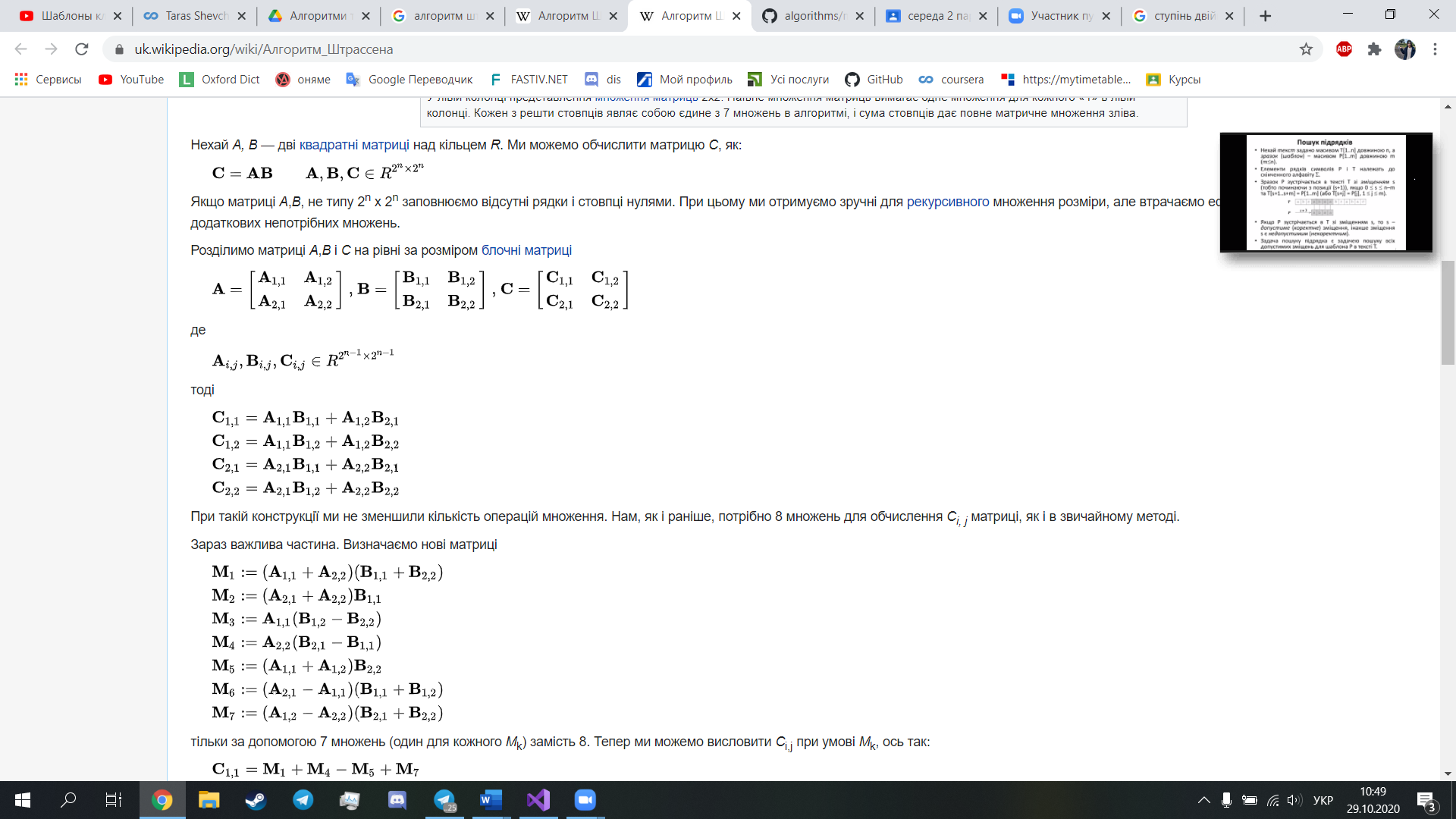
Реалізуйте алгоритм Штрассена для множення матриць. На практиці алгоритм починає застосовуватися для матриць такого розміру, коли з'являється виграш порівняно з класичним способом на основі означення, який використовується для матриць меншого розміру. Спробуйте експериментально визначити цю "точку перетину" для свого комп'ютера.

**Теорія**

Так як час роботи має бути лінійним використаємо префікс-функцію і те що, якщо рядок T є циклічним зсувом рядка Т\*, то рядок Т+Т буде містити в собі рядок Т\*.

**Алгоритм**

Маємо 2 квадратні матриці А та В. Матриця С є результатом перемноження цих матриць. Якщо розмір матриць, що множаться, не є натуральним степенем двійки, тоді доповнимо А та В нульовими рядками і стовпчиками. Розділимо матриці А та В на рівні блочні матриці.



Введемо наступні матриці:

𝑀1 = (A11 + A22) ∙ (B11 + B22)

𝑀2 = (A21 + A22) ∙ B11

𝑀3 = 𝐴11 ∙ B12 − B22

𝑀4 = 𝐴22 ∙ B21 − B11

𝑀5 = A11 + A12 ∙ 𝐵22

𝑀6 = A21 − A11 ∙ B11 + B12

𝑀7 = A12 − A22 ∙ B21 + B22

Тоді результуючі блоки обчислюються так:

𝐶11 = 𝑀1 + 𝑀4 − 𝑀5 + 𝑀7

𝐶12 = 𝑀3 + 𝑀5

𝐶21 = 𝑀2 + 𝑀4

𝐶22 = 𝑀1 − 𝑀2 + 𝑀3 + 𝑀6

Рекурсивно застосовуємо алгоритм, допоки розмір матриць не дійде до «переломного» рівня, тоді застосовуємо звичайний алгоритм множення

**Складність**

Звичайний алгоритм перемноження матриць виконується за О(n3). Використовуючи алгоритм Штрассена асимптотична складність для множення матриць розміром N = 2n буде О()

**Мова програмування**

С++

**Модулі програми**

* void MatrixMultiply(int A[][N], int B[][N], int R[][N])

множення матриць класичним способом

* void addition(int n, int A[][N], int B[][N], int R[][N])

додавання матриць

* void subtraction(int n, int A[][N], int B[][N], int R[][N])

віднімання матриць

* void Strassen\_algorithm(int n, int A[][N], int B[][N], int C[][N])

алгорим Штрассена

**Інтерфейс користувача**

Розмірність вводиться з консолі і результуюча матриця виводиться в консоль.

**Тестові приклади**

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **C** |
| **1 2 3**  **4 5 6**  **0 2 0** | **1 2 0**  **3 4 1**  **0 0 0** | **7 10 2**  **19 28 5**  **6 8 2** |
| **8 4 1 2**  **0 0 1 7**  **6 3 0 0**  **3 15 4 7** | **0 0 6 42**  **3 5 1 2**  **2 3 3 4**  **7 8 9 0** | **28 39 73 348**  **51 59 66 4**  **9 15 39 258**  **102 143 108 172** |

**Висновки**

На моєму комп’ютері алгоритм Штрассена ефективніше діє для матриць розміром від 128х128. На практиці його застосовують для суттєво більших матриць.

**Література**

* https://habr.com/ru/post/313258/
* https://studfile.net/preview/3068589/
* Лекція №7