Національний технічний університет України
"Київський політехнічний інститут ім. Ігоря Сікорського"

Навчально-науковий Фізико-технічний інститут
Кафедра математичних методів захисту інформації

КОМП'ЮТЕРНИЙ ПРАКТИКУМ 2: РЕАЛІЗАЦІЯ АЛГОРИТМУ ДИСКРЕТНОГО ЛОГАРИФМУВАННЯ СІЛЬВЕРА-ПОЛІГА-ГЕЛЛМАНА

Виконали студенти
групи ФІ-23
Чуй Тимофій і Малютіна Марина

Київ — 2025

Мета роботи

Ознайомлення з алгоритмом дискретного логарифмування Сільвера-Поліга-Геллмана. Практична реалізація цього алгоритму. Пошук переваг, недоліків та особливостей застосування даного алгоритму дискретного логарифмування. Практична оцінка складності роботи алгоритму.

Постановка задачі

1. Уважно прочитати методичні вказівки до виконання комп'ютерного практикуму.

2. Написати програму, що розв'язує задачу дисктерного логарифму шляхом звичайного перебору.

3. Написати програму, що реалізовує алгоритм Сільвера-Поліга-Геллмана для груп типу $\mathbb{Z}_p^*$

4. Застосовувати реалізований алгоритм Сільвера-Поліга-Геллмана та метод перебору до задач дискретного логарифма, які формує допоміжна програма, почергово зі збільшенням порядку p. У випадку, якщо допоміжна програма не справляється зі завданням генерації задачі, або реалізація студента не справляється з розв'язком задачі за відведений час, зупинити збільшення вхідного параметра.

5. Заміряти час роботи реалізації алгоритму Сільвера-Поліга-Геллмана та метод перебору для обох типів задач створених допоміжною програмою..

6. Порівняти результати часу роботи між методами і між типами задач. Створити візуалізацію залежності часу роботи від вхідного параметра. Пояснити результати.

Опис рішень

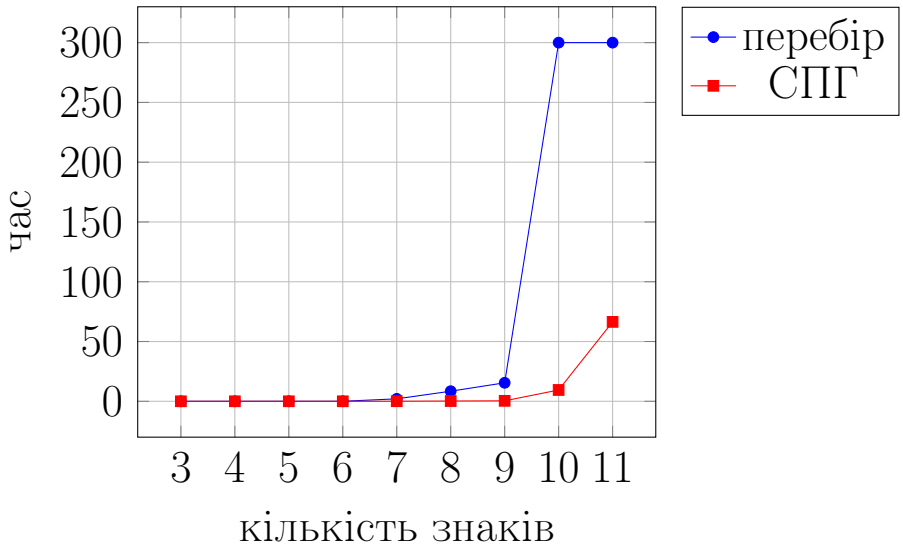
З методом перебору все очікувано просто, він працює по принципу "береш та порівнюєш". Щодо алгоритму Сільвера-Поліга-Геллмана, довелось трохи поламати голову над тим як простіше реалізувати, який тип даних використати для вхідних. В іншому - без суттєвих проблем.

Результати

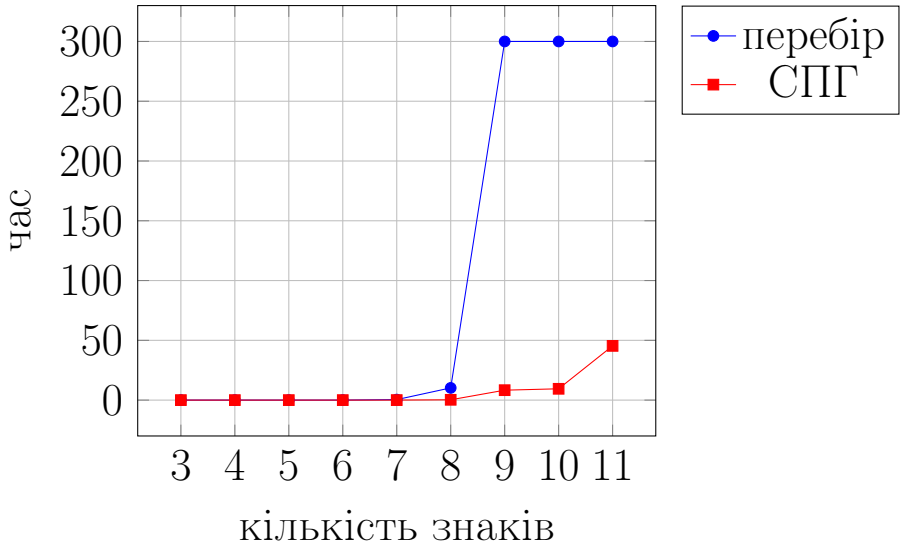
Нижче наведена таблиця з результатами заміру середнього часу 5 запусків кожного методу а також візуалізація, аби наглядно можна було побачити перевагу СПГ над перебором.

К-ть знаків	α	β	n	x	Час перебором	Час СПГ
3/тип 1	61	27	601	12	0.00005	0.0001
3/тип 2	126	561	719	200	0.0002	0.00004
4/тип 1	722	1789	2179	1206	0.0012	0.0001
4/тип 2	3048	932	5903	3841	0.0018	0.0003
5/тип 1	39105	46297	60289	38294	0.0158	0.0014
5/тип 2	26664	9606	31973	1266	0.0011	0.0001
6/тип 1	207764	25430	390869	40772	0.0203	0.0011
6/тип 2	69036	123995	521533	31270	0.0128	0.0012
7/тип 1	3676331	614793	4170541	3642800	2.0489	0.0581
7/тип 2	4131913	1763212	4877459	817961	0.4212	0.0138
8/тип 1	14390843	42225218	47487259	13798106	8.4382	0.2172
8/тип 2	32983050	56317687	83555669	16639597	10.2178	0.2650
9/тип 1	121563032	516170742	893533441	24637684	15.4651	0.3924
9/тип 2	474820254	40136582	937237349	528673215	300	8.3025
10/тип 1	933895063	1484143709	1639867079	603807270	300	9.4640
10/тип 2	3068468170	3940222237	7207030489	615948263	300	11.8292
11/тип 1	26708645486	19564897418	27574761517	1923177060	300	66.4284
11/тип 2	17435029047	10429220418	31788610771	14754109763	300	45.3546

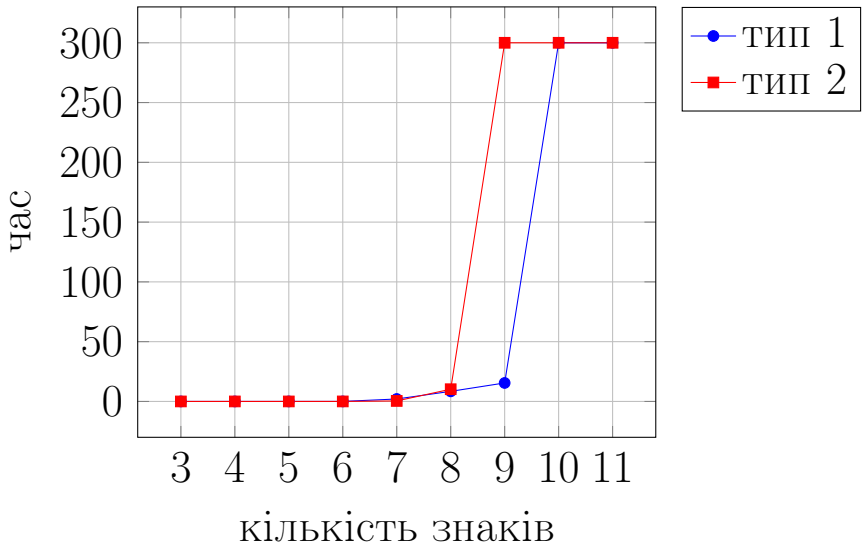
Задача типу один

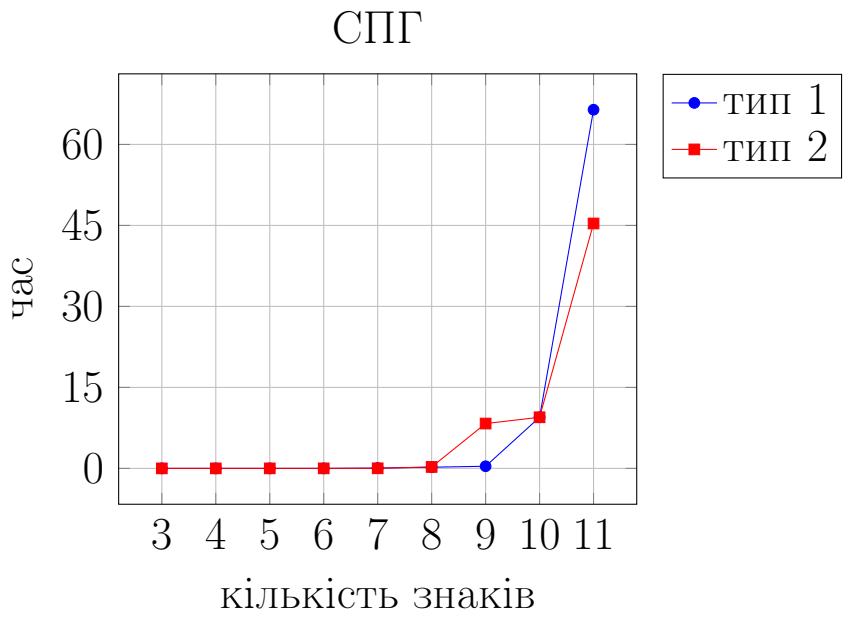


Задача типу два



Перебір





Висновки

Загальна картина така, що СПГ працює краще за перебір, були числа на яких перебір спрацював швидше, але це не заслуга методу, а просто удача, що розв'язок недалеко від 0. Розглядалась кількість знаків від 3 до 11, вбільших значеннях допоміжна програма не генерувала числа і наша реалізація виходила за рамки 5 хвилин. Із отриманих значень можна побачити як початкових значеннях перебір йшов майже в ногу з СПГ, а зі збільшенням довжини числа перебір першим вийшов за рамки 5 хвилин обчислень