**­**

**ЗВІТ**

**з дисципліни «Алгоритми та структури даних»**

**Лабораторне заняття № 7**

на тему: «Хеш таблиця закрите хешування»

студента групи ПТ-23-1

Звонарьов Віктор

1. **Мета роботи**

Вивчення методів представлення хеш-таблиці в пам'яті ЕОМ, а також освоєння засобів мови програмування для реалізації операцій з даними хеш-таблиці із закритим хешуванням.

1. **Умова завдання**

Реалізувати структуру хеш-таблиці. Реалізувати способи вирішення колізій при збігу значень хеш-функції:

Закрите хешування (відкрита адресація з лінійним пробінгом).

Реалізувати основні операції роботи з хеш-таблицями:

додавання нової пари «ключ-значення»;

видалення пари «ключ-значення» за ключем;

пошук значення за ключем.

1. **Структура основних вхідних та вихідних даних**

У програмі використовується клас HashTable, який містить такі поля:

- Node\*\* table — масив вказівників на пари «ключ-значення» (вузли).

- int size — розмір хеш-таблиці.

- int count — кількість елементів у хеш-таблиці.

- const int MIN\_SIZE — мінімально допустимий розмір таблиці для реалізації автоматичного зменшення розміру.

Основні функції для роботи з елементами таблиці

1. print() — виводить всі значення хеш-таблиці.

- Вхідні дані: void

- Вихідні дані: void

2. insert(int key, int value) — вставляє нову пару «ключ-значення» в хеш-таблицю.

- Вхідні дані: int key (ключ для хешування), int value (значення для зберігання)

- Вихідні дані: void

3. search(int key) — шукає значення за ключем в хеш-таблиці.

- Вхідні дані: int key (ключ для пошуку)

- Вихідні дані: int (значення, пов’язане з ключем, або -1, якщо ключ не знайдено)

4. remove(int key) — видаляє пару «ключ-значення» за заданим ключем.

- Вхідні дані: int key (ключ для видалення)

- Вихідні дані: bool (повертає true, якщо елемент було видалено, або false, якщо елемент з таким ключем не знайдено)

**4. Текст програми (листинг)**

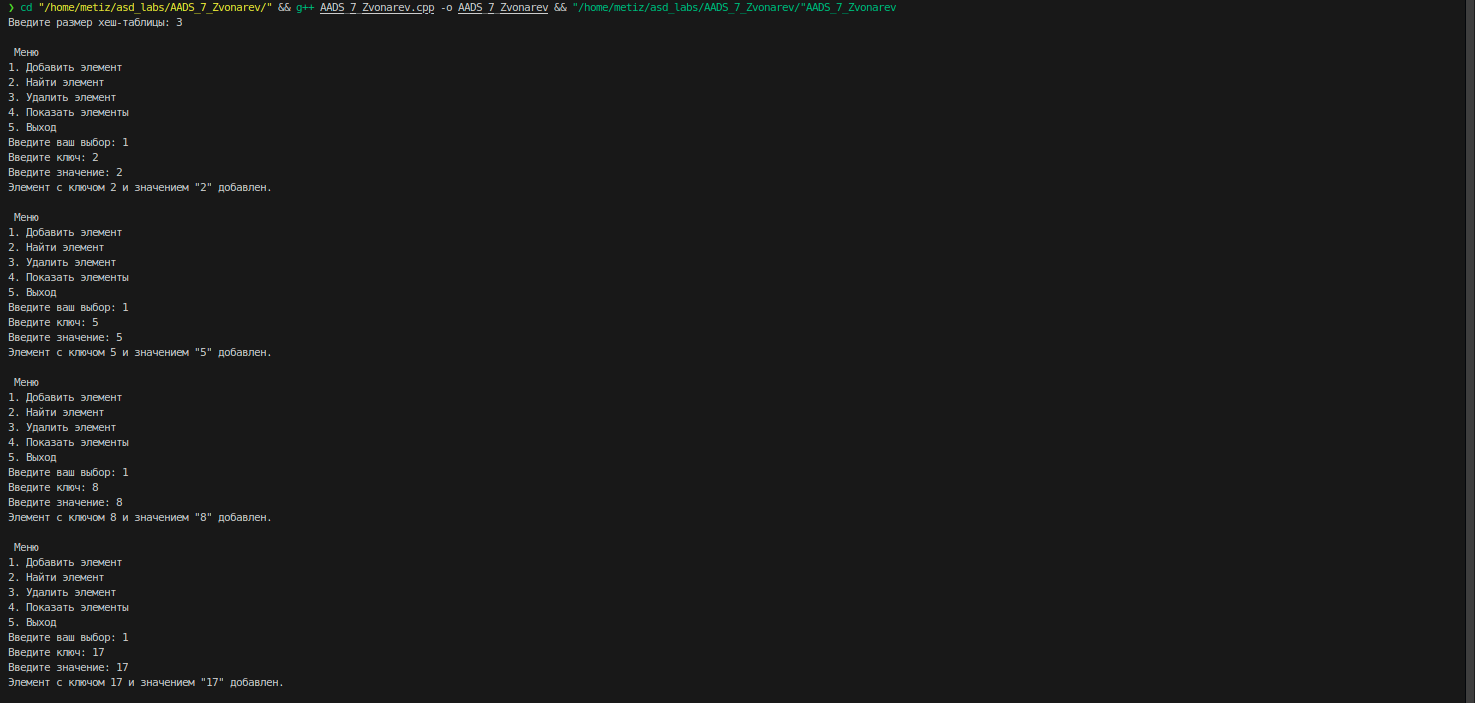
1. print() — проходить по кожному індексу таблиці та виводить елементи, що знаходяться на відповідних позиціях. Якщо позиція пуста або позначена як "видалена", виводиться "Empty".

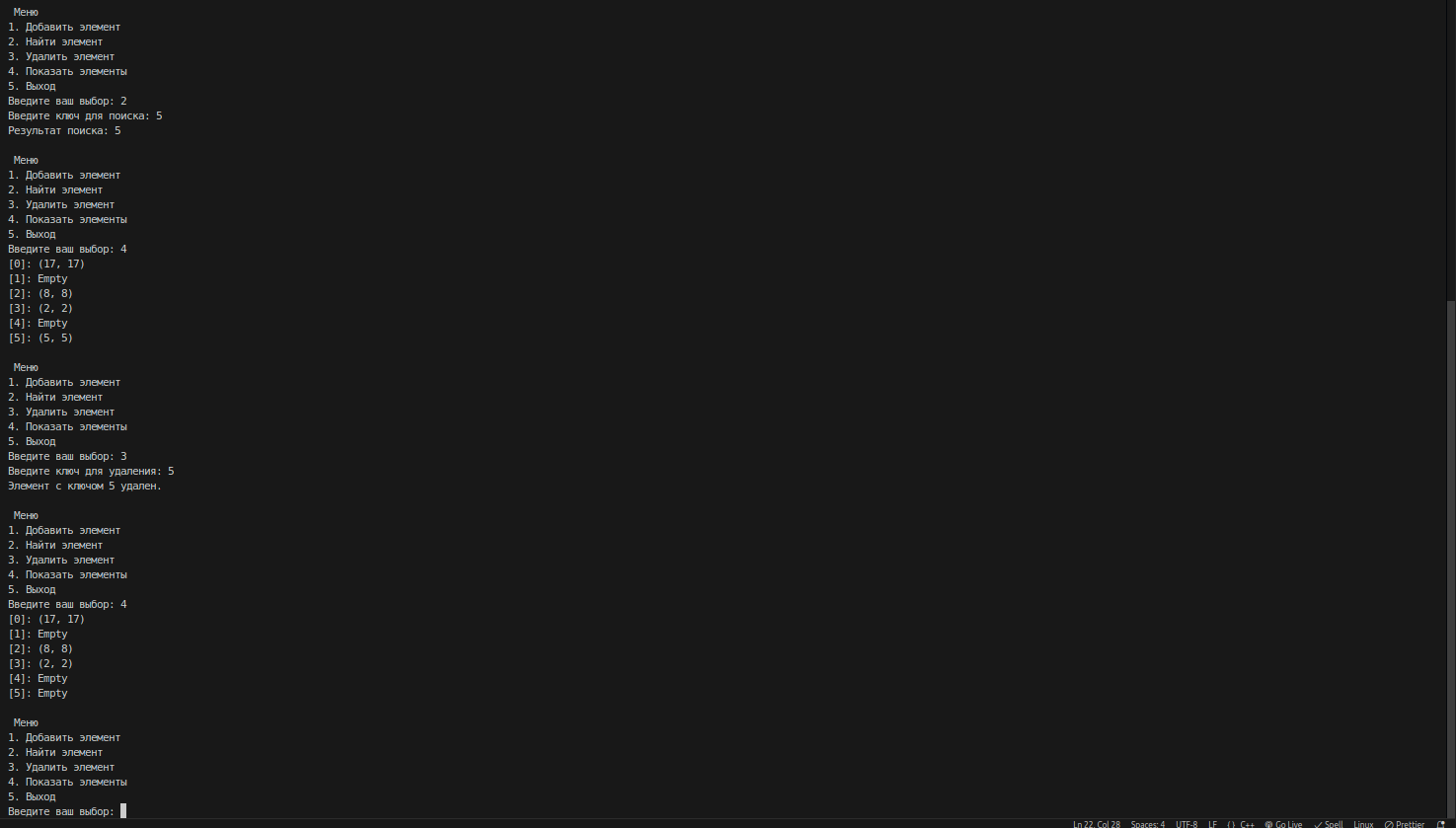
2. insert(int key, int value) — використовує хеш-функцію для обчислення індексу, куди слід вставити пару. Якщо місце зайняте іншим ключем, застосовується лінійний пробінг (перехід до наступного індексу) до знаходження вільної або видаленої позиції. Після додавання елемента програма перевіряє коефіцієнт завантаження таблиці і збільшує її розмір у разі перевищення 70%.

3. search(int key) — знаходить індекс елемента за хеш-функцією та виконує лінійний пробінг у разі колізій, поки не буде знайдено елемент із відповідним ключем або пусте місце. Якщо елемент знайдено, повертає його значення; якщо ні — повертає -1.

4. remove(int key) — визначає індекс за хеш-функцією і за необхідності виконує лінійний пробінг. При знаходженні потрібного елемента відзначає його як видалений. Якщо коефіцієнт завантаження зменшується нижче 50%, таблиця автоматично зменшується вдвічі (до мінімального значення).

**5. Набір тестів (Опис тестових прикладів)**

****

****

**6. Результати роботи програми та їх аналіз**

Для демонстрації роботи програми наведемо приклад взаємодії з хеш-таблицею із закритим хешуванням (відкрита адресація з лінійним пробінгом). У програмі передбачено функції для додавання, пошуку, видалення та відображення елементів хеш-таблиці.