МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ «ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «Програмна інженерія та інформаційні технології управління»

Звіт з лабораторної роботи №2

З предмету «Алгоритми та структури даних»

Виконав Студент групи КН-36а

Рубан Ю.Д.

Перевірила:

ас. Бородіна І. О.

Харків 2017

БАЗОВІ СТРУКТУРИ ДАНИХ. ХЕШ-ТАБЛИЦІ

Мета роботи: познайомитися з хеш-функціями та хеш-таблицями та отримати навички програмування алгоритмів, що їх обробляють.

Завдання

Розробити програму, яка читає з клавіатури цілі числа N, M (1 < N, M <

256), N пар <ключ, значення> (ключ — ціле, дійсне число або рядок в залежності від варіанту завдання; значення — рядок; усі рядки до 255 символів), жодний з яких не повторюється та ще M ключів. Всі рядки розділяються пробілом або новим рядком. Програма зберігає пар рядків до хеш-таблиці та видає на екран значення, що відповідають переліченим ключам.

Приклад **входу** для ключів-рядків.

- 3 2
- 11
- abc x
- gh yq
- io qw
- gh
- io

Вихід.

- yq
- qw

Використовувати готові реалізації структур даних (наприклад, STL) заборонено, але можна використати реалізацію рядків (наприклад, std::string у C++).

Варіант завдання

5 Ключ — рядок; хешування за остачею суми символів.

Хід виконання роботи

Розроблено програму, яка читає з клавіатури цілі числа N, M (1 < N, M < 256), N пар <ключ, значення> (ключ —рядок; значення — рядок; усі рядки до 255 символів), жодний з яких не повторюється та ще M ключів. Всі рядки розділяються пробілом або новим рядком. Програма зберігає пар рядків до хеш-таблиці та видає на екран значення, що відповідають переліченим ключам.

Код програми:

```
#include<iostream>
#include<string>
using namespace std;
class lList
{
private:
	lList *head = 0;
	lList *next = 0;
	lList *tail = 0;
	string data = "";
	string key = "";
	int count = 0;
public:
	void add(string d, string k)
	{
		lList *temp = new lList;
		temp->data = d;
		temp->key = k;
		if (head == NULL)
		{
			head = tail = temp;
		}
		else
		{
			tail->next = temp;
			tail = temp;
		}
		count++;
	}
	string getKeyViaIndex(int index)
	{
		lList*temp = head;
		if (index >= count) { throw; }
		for (int i = 0; i < index; i++)
		{
			temp = temp->next;
		}
		return temp->key;
	}
	string getElemViaIndex(int index)
	{
		lList*temp = head;
		if (index >= count) { throw; }
		for (int i = 0; i < index; i++)
		{
			temp = temp->next;
		}
		return temp->data;
	}
	int getCount() { return count; }
};
class Hash
{
public:
	Hash(int s)
	{
		size = s;
		table = new lList[size];
	}
```

```
        void add(string k, string w)
        {
                int index = hashCalc(k);
                table[index].add(w, k);
        }
        string search(string k)
        {
                int index = hashCalc(k);
                for (int i = 0; i < table[index].getCount(); i++)
                {
                        if (table[index].getKeyViaIndex(i) == k)
                        {
                                return table[index].getElemViaIndex(i);
                        }

                }
                return "there is no elem";
        }

private:
        int size;
        lList *table;
        int hashCalc(string k)
        {
                int length = k.length();
                int sum = 0;
                int *A = new int[length];
                for (int i = 0; i < length; i++)
                {
                        A[i] = (int)k[i];
                        sum += A[i];
                }
                sum = sum % size;
                return sum;
        }
};

int main()
{

        string key;
        string word;
        int m, n;
        cin >> n >> m;
        Hash table(n);
        for (int i = 0; i < n; i++)
        {
                cin >> key;
                cin >> word;
                table.add(key, word);
        }
        string sWord;
        for (int i = 0; i < m; i++)
        {
                cin >> sWord;
                cout << table.search(sWord) << endl;
        }
        system("pause");
        return 0;
}
```

Висновки

У даній лабораторній роботі я дізнався, що таке хеш-функція, дізнався як їх використовувати та навчився обробляти колізії методом ланцюгів.