МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

“ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”

КАФЕДРА ПРОГРАМНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ УПРАВЛІННЯ

ЗВІТ

З ЛАБОРАТОРНОЇ РОБОТИ №2

ЗА ТЕМОЮ “БАЗОВІ СТРУКТУРИ ДАНИХ. ХЕШТАБЛИЦІ”

Виконав студент

групи КН-221д

Габорець

Перевірив

Солонська С.В.

Харків 2021

ЛАБОРАТОРНА РОБОТА №2

Тема: БАЗОВІ СТРУКТУРИ ДАНИХ. ХЕШТАБЛИЦІ

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 хештаблиці та видає на екран значення, що відповідають переліченим ключам.

Варіант завдання:

Ключ — ціле число; хешування за допомогою ділення





Код:

#include <iostream>

using namespace std;

struct Link

{

string value = "";

Link\* previous = nullptr;

~Link() {};

};

struct HashTable

{

int key = 0;

//string value = "";

Link\* tail = nullptr;

};

int hash\_function(int key, int size)

{

return key % size;

}

void init\_array(struct HashTable\* array, int size)

{

for (int i = 0; i < size; i++)

{

array[i].key = 0;

//array[i].value = "";

array[i].tail = new Link; //указатель tail указывает на узел(Линк) с Данными

//array[i].tail->value = "";

}

}

void add\_element(struct HashTable\* array, int key, string value, int size)//Заполняем

{

int index = hash\_function(key, size);

//if (array[index].value == "")

if (array[index].tail->value == "")

{

array[index].key = key;//Пихаем туда ключ

//array[index].value = value;

array[index].tail->value = value;//Пихаем туда Значение под этим ключом

//size++;

cout << "\nKey " << key << " was edited\n";

}

//else if (array[index].key == key)

//{

//array[index].value = value;

//array[index].tail.value = value;

//}

else

{

//cout << "\nCollision was happaned\n";

Link\* link = new Link;//Тута если у нас получилась (коллизия - одинаковый хеш)

link->previous = array[index].tail;

array[index].tail = link;

array[index].tail->value = value;

}

}

/\*void remove\_element(struct HashTable\* array, int key, int size)

{

int index = hash\_function(key, size);

{

cout << "\nThis key does not exist\n";

}

else

{

array[index].key = 0;

//array[index].value = "";

array[index].tail->value = "";

cout << "\nKey " << key << " was deleted\n";

}

}\*/

int main()

{

unsigned int N = 0, M = 0;

while (N <= 1 || N >= 256 || M <= 1 || M >= 256)

{

cout << "Input 1 < N < 256:";

cin >> N;

cout << "Input 1 < M < 256:";

cin >> M;

}

unsigned int size = N + M;

struct HashTable\* array = new HashTable[size];

init\_array(array, size);

cout << "Input key/value pairs:\n";

for (unsigned int i = 0; i < N; i++)

{

int key;

cin >> key;

string value;

cin >> value;

add\_element(array, key, value, size);

}

cout << "Input ONLY keys(M):\n";

for (unsigned i = 0; i < M; i++)

{

int key;

cin >> key;

add\_element(array, key, "", size);

}

int choose = 1;

while (choose == 1)

{

cout << "\nInput key and get key/value pairs:\n";

int key;

cin >> key;

Link\* link = array[hash\_function(key, size)].tail;

//cout << "\t" << array[hash\_function(key, size)].key << "\t" << array[hash\_function(key, size)].value << endl;

/\*cout << "\n\n" << "your key: " << array[hash\_function(key, size)].key << endl;\*/

cout << "your value:" << link->value;

while (link->previous != nullptr)//с хвоста идёт

{

link = link->previous;

cout << "\t" << link->value;

}

cout << endl;

std::cout << "Do you want to continue?\nIf yes choose 1 , if no choose 2!\n";

cin >> choose;

}

}

Висновок: виконавши лабораторну роботу №2, я познайомився з базовими структурами даних (список, черга, стек) та отримав навички програмування алгоритмів, що їх обробляють.