Міністерство освіти і науки України

Національний Технічний Університет

«Харківський Політехнічний Інститут»

Кафедра «Стратегічного Керування»

Лабораторна робота № 2

«ХЕШ-ТАБЛИЦІ»

Перевірила: Мошко Є. О.

Виконав:

Харків, 2018

Лабораторна робота № 2

Тема: ХЕШ-ТАБЛИЦІ

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 тавидає на екран значення, що відповідають переліченим ключам.

Варіант завдань

2 варіант. Ключ —ціле число; хешування за допомогою ділення.

Хід роботи

|  |
| --- |
| #include <iostream>  using std::cout;  using std::cin;  using std::endl;  using std::stringstream;  const int TABLE\_SIZE = 10;  class HashMap{  private:  int\*\* arr;  public:  HashMap(){  arr = new int\* [TABLE\_SIZE];  for (int i=0; i<TABLE\_SIZE;i++){  arr[i]=NULL;  }  }  void put(int key, int value){  int hash=key%TABLE\_SIZE;  if (arr[hash]==NULL){  arr[hash] = new int[TABLE\_SIZE];  for (int i=0; i<TABLE\_SIZE;i++){  arr[hash][i]=NULL;  }  arr[hash][0]=value;  }  else  {  bool lock;  int i=0;  for (i; arr[hash][i]!=NULL; i++){  if (arr[hash][i]==value) {  lock=true;  cout << "Value is already in the key" << endl;  break;  }  }  if (!lock){  arr[hash][i]=value;  }  }  }  int\* get(int key){  int hash=key%TABLE\_SIZE;  return arr[hash];  }  };  int main() {  int key, value;  int N, M;  int srch;  cout << "Enter amount of pairs & amounts of keys:" << endl;  cin >> N >> M;  cout << "Enter "<< N << " pairs (key value):" << endl;  HashMap mymap;  for (int i=0; i<N; i++){  cin >> key >> value;  mymap.put(key,value);  }  cout << "Enter " << M << " keys: " << endl;  for (int j=0; j<M;j++){  cin >> key;  int\* arr = mymap.get(key);  if (arr){  for (int i=0;arr[i]!=NULL;i++){  cout << arr[i] << " ";  }  }  else  {  cout << "None";  }  cout << endl;  }  system ("pause");  } |

Приклад виконання програми:

|  |
| --- |
| Enter amount of pairs & amounts of keys:  4 4  Enter 4 pairs (key value):  1 3  1 5  2 5  3 65  Enter 4 keys:  1  3 5  3  65  5  None  6  None  Press any key to continue . . . |

Висновок

Я ознайомився з хешуванням – перетворенням будь-якого розміру в дані фіксованого розміру за допомогою хеш-функціі. Хеш-функції використовуються для оптимізації таблиць та баз даних за рахунок того, що у однакових записів однакові значення хеш-функції. Існує безліч масивів з різним вмістом, що дають однакові хеш-коди — так звані колізії. Імовірність виникнення колізій відіграє важливу роль в оцінці якості хеш-функцій. Для боротьби з колізіями у цьому коді було використано «метод ланцюжків» коли визначається масив вказівників на зв’язні списки.