МИНИСТЕРСТВО ОБРАЗОВАНИЯ и НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Коломенский институт (филиал)**

федерального государственного бюджетного образовательного учреждения

высшего образования

**«Московский политехнический университет»**

Профиль «**Программное обеспечение вычислительной техники и автоматизированных систем**»

**Направление подготовки:**

**«Информатика и вычислительная техника»**

Форма обучения: очная

**лабораторная работа № 8**

по дисциплине «Структуры и алгоритмы обработки данных»

Выполнил студент группы ИВТ-21 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /М.А. Ковалёва

(подпись) И.О. Фамилия

Проверил преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ И.Н. Филоненко /

(подпись) И.О. Фамилия

Дата:

Коломна 2020

**Лабораторная работа № 8**

**Хеширование. Алгоритмы организации и обработки хеш-таблиц.**

Построить хеш-таблицу с открытой адресацией, используя двойное хеширование, как способ открытой адресации. Хеш-функция h1(k) и h2(k) организовать методом деления. Формирование таблицы - с помощью процедуры поиска и вставки по ключу.

unit Unit1;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, Vcl.Grids;

type

TForm1 = class(TForm)

Label1: TLabel;

StringGrid1: TStringGrid;

Edit1: TEdit;

Button1: TButton;

Label2: TLabel;

procedure FormShow(Sender: TObject);

procedure insertHash(key: integer);

procedure tableprint();

procedure Button1Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

Const

{$WriteableConst On}

tablesize: integer = 20;

dhfvalue: integer = 17;

var

Form1: TForm1;

fullChecker: integer;

hashtable: array[1..20] of integer;

implementation

{$R \*.dfm}

function isFull():boolean;

begin

if fullchecker = tablesize then

isFull := True

else

isFull := False;

end;

function h1(key: integer): integer;

begin

h1:= key mod tablesize;

end;

function h2(key: integer): integer;

begin

h2 := dhfvalue - (key mod dhfvalue);

end;

procedure initTab();

var

i: integer;

begin

fullChecker := 1;

for i := 1 to tablesize do

hashtable[i] := -1;

end;

procedure TForm1.insertHash(key: integer);

var

index, index2, i, newindex:integer;

begin

if isFull() = True then

showmessage('Таблица заполнена!' + ' ' + inttostr(fullchecker) + ' ' + inttostr(tablesize))

else begin

index := h1(key);

if hashtable[index] <> -1 then

begin

index2 := h2(key);

i := 1;

while True do

begin

newindex := (index + i \* index2) mod tablesize;

if hashtable[newindex] = -1 then

begin

hashtable[newindex] := key;

break;

end;

inc(i);

end;

end

else

hashtable[index] := key;

inc(fullChecker);

end;

end;

procedure TForm1.tableprint();

var

i: integer;

begin

for i := 1 to tablesize do

if hashtable[i] = -1 then

StringGrid1.Cells[i - 1, 1] := ''

else

StringGrid1.Cells[i - 1, 1] := inttostr(hashtable[i]);

end;

procedure TForm1.Button1Click(Sender: TObject);

var

addvalue: integer;

begin

addvalue := strtoint(Edit1.Text);

insertHash(addvalue);

tableprint();

end;

procedure TForm1.FormShow(Sender: TObject);

var

i: integer;

begin

Randomize;

initTab();

StringGrid1.ColCount := tablesize;

for i := 1 to tablesize do

begin

stringGrid1.Cells[i - 1, 0] := inttostr(i);

end;

for i := 1 to 10 do

begin

insertHash(1 + Random(100));

end;

tableprint();

end;

end.



