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 № 6**

по дисциплине «Структуры и алгоритмы обработки данных»

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

(подпись) И.О. Фамилия

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

(подпись) И.О. Фамилия

Дата:

Коломна 2020

**Задание:**

|  |  |
| --- | --- |
| 1. | а )Организовать и отладить программу для построения и печати идеально сбалансированного двоичного дерева ( Function Tree , Function PrintTree ) .  б )Запрограммировать и отладить алгоритм обхода построенного бинарного дерева слева направо (в качестве примера построить и обойти дерево , отображающее заданное арифметическое выражение с бинарными операциями). |

**Код:**

type

ptr=^node; //ptr типизированый указатель

node=record //запись данных

left,right:ptr; //указатели лево/право

key:integer; //ключ целый

end;

var

Form2: TForm2;

implementation

{$R \*.dfm}

procedure TForm2.Button1Click(Sender: TObject);

var

po:ptr;

i,k,t:integer;

a:array of integer;

function Tree(n:integer):ptr; //строит дерево

var

newnode:ptr;

nl,nr:integer;

begin

if n=0 then //если количество элементов 0

newnode:=nil

else begin

nl:=n div 2;

nr:=n-nl-1;

getmem(newnode,sizeof(node)); //выделение памяти под элемент

newnode^.key:=a[t]; //элемент массива

inc(t); //t+1

newnode^.left:=tree(nl); //рекурсия

newnode^.right:=tree(nr); //рекурсия

end;

tree:=newnode; //вывод ссылочной величины newnode

end;

Procedure inorder(t:ptr);

begin

if t<>nil then begin

inorder(t^.left);

memo2.lines.add(inttostr(t^.key));

inorder(t^.right);

end;

end;

function printTree (Treenode:Ttreenode; t:ptr); // Treenode:Ttreenode - узел

var

newnode:TtreeNode;

begin

if assigned(t) then begin //является ли указатель nil.

newnode:= TreeView1.Items.AddChild(treenode, IntToStr(t^.key));

//Добавляет узел с ключем как последний дочерний узла Treenode.}

printTree(newnode,t^.left); // левое

printTree(newnode,t^.right); //правое

end;

end;

begin

k:=memo1.lines.count;

setlength(a,k);

for i:=0 to k-1 do

a[i]:=strtoint(memo1.Lines[i]);

t:=0;

po:=tree(k);

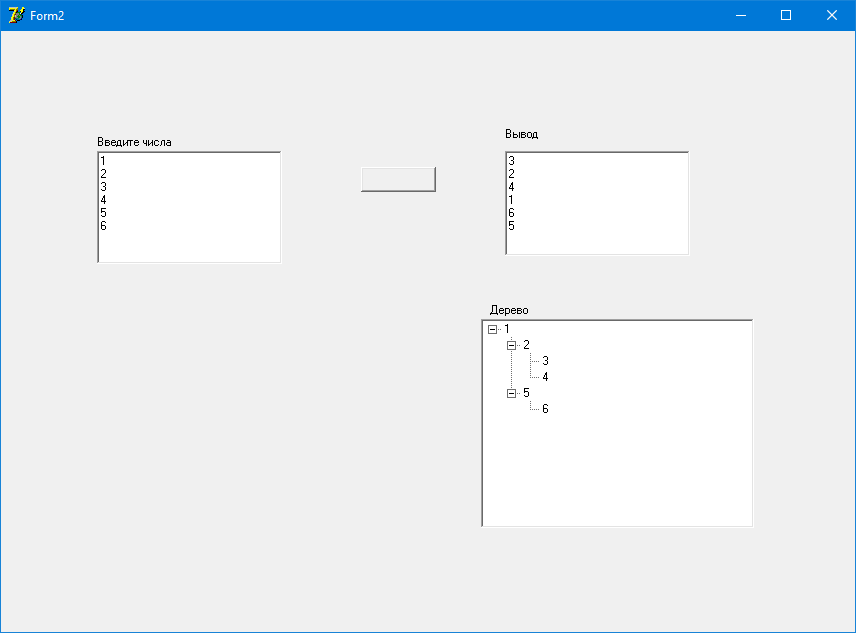
inorder(po);

printTree(nil,po);

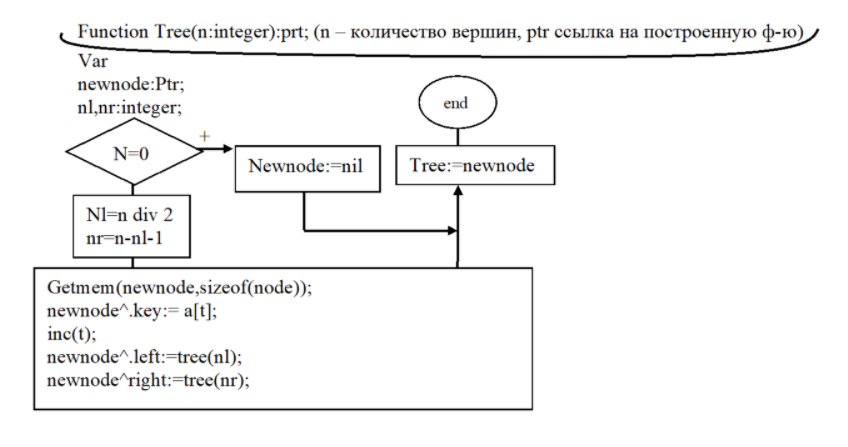
end;

end.

**Результат:**



**Блок схема:**



­