Лабораторная работа выполнется на языках высокого уровня (Паскаль, Си, Delphi, Builder C++, Visual C++. Необходимо представить

* Исходные тексты программ с подробными комментариями;
* Исполняемые файлы;
* Отчет по лабораторной работе.

Отчет должен включать в себя следующие разделы

* Формулировку задания
* Описание основных методов, используемых в лабораторной работе;
* Результаты работы программы (в виде файла или в виде скриншота);
* Анализ результатов.

Тестирование программ должно проводиться для различных случаев: упорядоченный массив (прямой или обратный порядок), случайный массив.

**Задание**

Тема: Построение случайного дерева поиска и идеально сбалансированного дерева поиска

Цель работы: Освоить методы построения случайного дерева поиска и идеально сбалансированного дерева поиска.

Порядок выполнения работы:

1. Разработать процедуры построения СДП и ИСДП.
2. Вычислить среднюю высоту построенных деревьев для *n*=10, 50, 100, 200, 400 (*n* -количество вершин в дереве). Заполнить таблицу следующего вида и проанализировать полученные результаты

|  |  |  |
| --- | --- | --- |
| *n*  | Высота СДП  | Высота ИСДП  |
| 10  |  |  |
| 50  |  |  |
| 100  |  |  |
| 200  |  |  |
| 400  |  |  |

1. Написать процедуру, определяющую является ли двоичное дерево деревом поиска. Проверить ее работу на построенных СДП и ИСДП.
2. Запрограммировать процедуру поиска в дереве поиска элемента с заданным ключом и проверить ее работу на построенных СДП и ИСДП.
3. Определить количество операций, необходимых для поиска. Сравнить эту величину с высотой дерева.

**Приложение**

**Псевдокод для записи алгоритмов**

Для записи алгоритма будем использовать специальный язык - псевдокод. Алгоритм на псевдокоде записывается на естественном языке с использованием двух конструкций: ветвления и повтора. В круглых скобках будем писать комментарии. В треугольных скобках будем описывать действия, алгоритм выполнения которых не требует детализации, например, <обнулить массив>.

: = Операция присваивания значений.

Операция обмена значениями.

Конструкции ветвления.

IF (условие) Если выполняется условие,

<действие> то выполнить действие

FI FI указывает на конец этих действий.

IF (условие)

<действия 1>

ELSE <действия 2> Действия 2 выполняются,

FI если неверно условие.

IF (условие1)

<действия1>

ELSEIF (условие2) Действия 2 выполняются,

<действия2> если неверно условие1 и верно условие 2

...FI

Конструкции повтора.

Цикл с предусловием.

DO (условие) Действия повторяются

<действия> пока условие истинно.

OD OD указывает на конец цикла.

Цикл с постусловием.

DO <действия>

OD (условие выполнения)

Цикл с параметром.

DO (i=1, 2, ... n) Действия выполняются для значений

<действия> параметра из списка

OD

Бесконечный цикл.

DO

<действия>

OD

Принудительный выход из цикла.

DO

...IF (условие) OD Если условие истинно, то выйти из цикла.

OD