Варианты заданий для

лабораторной работы

на тему «Поиск»

Для выполнения лабораторной работы необходимо **сгенерировать** тестовые файлы (используя генераторы случайных чисел), содержащие целые числа, в количестве от 26 до 216 (можно и больше), при этом количество элементов в следующем файле в два раза больше чем в предыдущем.

Для **каждого** тестового файла из набора выполнить следующие действия:

1. Поиск элементов, которые гарантированно имеются во входных данных.

Выполнить поиск каждого элемента. Вычислить среднее время поиска по *одному* файлу.

1. Поиск элементов, которые гарантированно не имеются в исходных данных.

Вычислить среднее время поиска по *одному* файлу.

Построить **график** зависимости времени поиска от количества элементов в файле.

**Проверить** существование исходных данных таких, что время работы алгоритма сильно увеличивается по сравнению со временем работы алгоритма на случайных данных.

Выполнить **сравнение** алгоритмов из варианта:

1. Сравнить графики,
2. Сравнить асимптотические оценки.

**Отчет** должен содержать:

1. Код генератора исходных данных и данных для поиска,
2. Описание алгоритмов:
3. математические основы,
4. псевдокод.
5. Сравнительный анализ алгоритмов,
6. Программный код (необязательно),
7. Вывод.

***Количество баллов за выполнение вариантов 1-3 15.6 из26***

**Вариант 1**

-последовательный поиск по неупорядоченному файлу;

- бинарный поиск.

Перед просчетом упорядочить файл по

возрастанию ключей. Время сортировки не включается в измеряемое время.

- поиск с помощью стандартного алгоритма в стандартном контейнере.

**Вариант 2**

**-** бинарный поиск.

Перед просчетом упорядочить файл по

возрастанию ключей. Время сортировки не включается в измеряемое время.

- поиск по бинарному дереву поиска

**Вариант 3**- интерполяционный поиск  
- поиск по бинарному дереву поиска.

***Количество баллов за выполнение вариантов 4-10 ≥ 20.8 из26***

**Вариант 4**- бинарный поиск  
- поиск в *Splay*дереве

**Вариант 5**- бинарный поиск  
- поиск в *Red-Black*дереве

**Вариант 6**- бинарный поиск  
- поиск в *B* дереве

**Вариант 7**- бинарный поиск  
- поиск в *AVL* дереве

**Вариант 8**- поиск в хэш – таблице (способ разрешения коллизий – метод цепочек)  
- бинарный поиск

**Вариант 9** - поиск в хэш – таблице (способ разрешения коллизий – бинарное дерево)- бинарный поиск

**Вариант 10** - поиск в *Декартовом* дереве- бинарный поиск

***Количество баллов за выполнение вариантов 11-15 ≥ 20.8 из26 (менее требовательны на защите)***

**Вариант 11**- поиск в *Red-Black* дереве  
- поиск в *Splay* дереве

**Вариант 12**- поиск в *Red-Black* дереве  
- поиск в *B* дереве

**Вариант 13**- поиск в *AVL* дереве  
- поиск в *Splay* дереве

**Вариант 14**- поиск в *B* дереве  
- поиск в *AVL* дереве

**Вариант 15**  
 - поиск в *Декартовом* дереве  
- поиск в *Red-Black* дереве