**Операционные системы и системное программирование**

**Лабораторная работа № 4**

4."Телефонный справочник"

Существует телефонный справочник Минска в текстовом формате:

Телефон, Фамилия, Имя, Отчество, Улица, Дом, Корпус, Квартира.

При запуске программы справочник полностью загружается в память. На основе загруженного справочника сформировать индексную структуру для

эффективного поиска по определенным полям справочника (например: Фамилия, Телефон, Улица).

Расположить ядро БД (функции по загрузке, индексации, поиску и т.д.) в DLL-библиотеке. DLL-библиотека динамически загружается при запуске

основной графической программы. Программа-справочник должна быть реализована в виде приложения с графическим интерфесом, написанном на C++.

Предусмотреть возможность работы нескольких копий программы с одной базой данных.

Размер справочника и индекса должен помещаться в ОЗУ. Например, справочник размером 1-2Mb.

Задание на 8 баллов:

Загрузить файл в память, но организовать поиск данных, используя индекс. Индекс строится при загрузке текстовой базы данных. Индексации подлежат минимум 3 поля, например: Фамилия, Улица, Телефон.

Варианты структур:

1) бинарное дерево поиска (Binary Search Tree);

2) хеш-таблица (Hash-Table) с односвязным списком;

3) B-дерево (B-Tree);

4) преффиксное дерево (Trie)

5) LSM дерево

6) Список с пропусками (Skip List)

7) AVL дерево (AVL tree)

8) красно-черное дерево (Red–black tree)

Задание на 10 баллов:

1) Реализовать параллельный поиск записей.

2) Реализовать возможность редактирования данных.

Программу можно реализовывать в парах - графическая часть + DLL с БД.

Проект должен разрабатываться с использование системы контроля версий git/svn (рекомендуется git).

Все функции/процедуры, большие блоки кода должны содержать комментарии. Язык комментариев - английский.

--------------------------------------------

№ варианта |№ задания на 8 |№ задания на 10|

--------------------------------------------

1 | 1 | 1 |

--------------------------------------------

2 | 2 | 1 |

--------------------------------------------

3 | 3 | 1 |

--------------------------------------------

4 | 4 | 1 |

--------------------------------------------

5 | 5 | 1 |

--------------------------------------------

6 | 6 | 1 |

--------------------------------------------

7 | 7 | 1 |

--------------------------------------------

8 | 8 | 1 |

--------------------------------------------

9 | 1 | 2 |

--------------------------------------------

10 | 2 | 2 |

--------------------------------------------

11 | 3 | 2 |

--------------------------------------------

12 | 4 | 2 |

--------------------------------------------

13 | 5 | 2 |

--------------------------------------------

14 | 6 | 2 |

--------------------------------------------

15 | 7 | 2 |

--------------------------------------------

16 | 8 | 2 |

--------------------------------------------