**КР Раб.**

Присылаемый на проверку архив должен содержать 3 файла:

* файл отчета, содержащий исходный текст программы и результаты работы программы (можно в виде скриншотов);
* файл с исходным текстом программы на языке SWI-PROLOG;
* файл базы данных, с которым работает программа (не менее 10 строк).

**Задание:**

Напишите на языке SWI-PROLOG программу для работы с базой данных по заданию. Начальная база данных должна храниться в файле. В программе должно присутствовать меню из 5 пунктов, реализующих следующие возможности:

1. просмотр содержимого динамической базы данных (при запуске программы динамическая база данных должна заполняться из файла базы данных только один раз и только в этом пункте);

2. добавления записи в динамическую базу данных (за один вход в этот пункт должна быть возможность добавления нескольких записей);

3. удаления записи из динамической базы данных (за один вход в этот пункт должна быть возможность удаления нескольких записей);

4. выполнения запроса к динамической базе данных по заданию;

5. выход из программы с сохранением содержимого динамической базы данных в исходный файл базы данных (сохранение должно быть только один раз и только в этом пункте).

Создайте базу данных городского транспорта: название транспорта, номер маршрута, список остановок. Определите, на каких маршрутах можно добраться от одной остановки до другой без пересадок. Названия остановок вводятся с клавиатуры.

**Лаб Раб**

Присылаемый на проверку архив должен содержать 2 файла:

файл отчета исходный текст программы и результаты работы программы (можно в виде скриншотов);

файл с исходным текстом программы на языке CLISP.

**Задание**

Напишите на языке CLISP программу для работы со списками по заданию. Обязательно использование рекурсии. Ввод всех входных данных должен запрашиваться с клавиатуры в процессе работы программы (функции). Не допускается использование: функционалов, а в теле рекурсивной функции - операторов SET и SETQ.

Преобразуйте одноуровневый список L во вложенный по следующему правилу. Например, при L=(a s d f g) функция должна вернуть (a (s (d (f (g))))).

**№2**

**Обработка списков в языках CLISP и SWI-PROLOG**

файл отчета, исходные тексты программ и результаты работы программ (можно в виде скриншотов);

файл с исходным текстом программы на языке CLISP;

файл с исходным текстом программы на языке SWI-PROLOG.

**Задание**

Напишите на двух языках CLISP и SWI-PROLOG программы для работы со списками по заданию. Обязательно использование рекурсии. Ввод всех входных данных должен запрашиваться с клавиатуры в процессе работы программы (функции или предиката). В программе на языке CLISP не допускается использование: функционалов, а в теле рекурсивной функции - операторов SET и SETQ. В программе на SWI-PROLOG предикат после вывода результатов работы должен успешно завершиться (на экране появится true. и приглашение следующего ввода цели ?-).

Номер варианта выбирается по последней цифре пароля.

**1.** Удалите из списка L все вхождения элемента X.

Например, для языка CLISP при L=(1 2 1 3 1 1 2 5 6), X=1 функция должна вернуть (2 3 2 5 6).

**№3**

**Работа со строками и файлами в языке SWI-PROLOG**

Номер задачи выбирается по последней цифре пароля.

Присылаемый на проверку архив должен содержать 3 файла:

файл отчета исходный текст программы и результаты работы программы (можно в виде скриншотов);

файл с исходным текстом программы на языке SWI-PROLOG;

файл cо входными данными для программы.

**Задание**

Напишите на языке SWI-PROLOG программу для работы с текстовым файлом, состоящим из нескольких строк. При преобразовании файла должны быть сохранены переводы строк исходного файла.

Номер варианта выбирается по последней цифре пароля.

1. Запишите в новый файл все строки исходного файла, содержащие в качестве фрагмента заданную строку, которая вводится с клавиатуры.