**Задание Б – «СПИСОК»**

**Часть Б1. Смоделировать двумя способами** новый тип «Список»:

1) на основе одномерного динамического массива;

А

Б

В

Г

Д

Запас1 – при удалении не выгодно каждый раз сдвигать все элементы в начало массива, эффективнее накопить сначала M «свободных» элементов.

Конец  
(индекс)

Начало  
(индекс)

Запас2 – при добавлении не выгодно каждый раз выделять новую память и копировать в нее массив, эффективнее выделять память с запасом M элементов.

Для этого **создать модуль** и описать **новый тип данных** и его структуру (на русском языке и на Delphi/C, без классов! Только записи/структуры, массивы и файловый тип) и **базовые операции**:

- проверка на пустоту списка;

- создание пустого списка или списка из одного элемента (как удобнее);

- добавление элемента в список (в начало/в конец/после текущего);

- удаление элемента из списка (из начала/из конца/после текущего);

- сделать текущим элементом в списке (первый/ последний/ следующий/ предыдущий);

- считать значение текущего элемента из списка;

- вывод (дописывание) содержимого списка в текстовый лог-файл для контроля за состоянием списка после каждой операции с начала и до конца работы программы.

**Часть Б2. Создать приложение** (консольное, на C или Delphi), позволяющее выбрать и выполнить следующие действия над списком (только с помощью выше перечисленных базовых операций! Без прямого обращения к элементам массива/файла):

- очистить список;

- добавить элементы из текстового файла (в начало/ в конец/ после текущего);

- добавить один элемент, на основе данных введенных в интерактивном режиме (с клавиатуры или с формы) (в начало/ в конец/ после текущего);

- удалить N элементов списка (из начала/ из конца/ после текущего);

- сделать текущим элементом первый/последний/предыдущий/следующий;

- фильтр: *удалить* из списка все элементы, не удовлетворяющие *условию поиска*, указанному в списке задач по вариантам (стр.3), и *продублировать* удовлетворяющие условию.

После каждой операции выводить содержимое списка в лог-файл.

При *добавлении* элемента предусмотреть его *коррекцию* в соответствии с условием *коррекции* из списка задач по вариантам (стр.3).

19. *Исходные данные*: О детях, поступающих в спортивную секцию, собраны следующие данные: фамилия, имя, год рождения, вес, рост. *Условие поиска*: Дети от 10 до 12 лет, рост которых не менее 155см, а вес не более 45 кг. *Коррекция*: Все дети не младше 5 лет и не старше 14 по году рождения, рост-вес положительные и не более 200.

З**адание Б – «СПИСОК»**

**Часть Б1. Смоделировать двумя способами** новый тип «Список»:

2) на основе типизированного/двоичного файла.

А

Б

В

Г

Д

Е

Ё

Запас – при удалении не выгодно каждый раз сдвигать все элементы в начало файла, эффективнее накопить сначала M «свободных» элементов.

Конец  
(позиция в файле)

Начало  
(позиция в файле)

Ж

Для этого **создать модуль** и описать **новый тип данных** и его структуру (на русском языке и на Delphi/C, без классов! Только записи/структуры, массивы и файловый тип) и **базовые операции**:

- проверка на пустоту списка;

- создание пустого списка или списка из одного элемента (как удобнее);

- добавление элемента в список (в начало/в конец/после текущего);

- удаление элемента из списка (из начала/из конца/после текущего);

- сделать текущим элементом в списке (первый/ последний/ следующий/ предыдущий);

- считать значение текущего элемента из списка;

- вывод (дописывание) содержимого списка в текстовый лог-файл для контроля за состоянием списка после каждой операции с начала и до конца работы программы.

**Часть Б2. Создать приложение** (консольное, на C или Delphi), позволяющее выбрать и выполнить следующие действия над списком (только с помощью выше перечисленных базовых операций! Без прямого обращения к элементам массива/файла):

- очистить список;

- добавить элементы из текстового файла (в начало/ в конец/ после текущего);

- добавить один элемент, на основе данных введенных в интерактивном режиме (с клавиатуры или с формы) (в начало/ в конец/ после текущего);

- удалить N элементов списка (из начала/ из конца/ после текущего);

- сделать текущим элементом первый/последний/предыдущий/следующий;

- фильтр: *удалить* из списка все элементы, не удовлетворяющие *условию поиска*, указанному в списке задач по вариантам (стр.3), и *продублировать* удовлетворяющие условию.

После каждой операции выводить содержимое списка в лог-файл.

При *добавлении* элемента предусмотреть его *коррекцию* в соответствии с условием *коррекции* из списка задач по вариантам (стр.3).

19. *Исходные данные*: О детях, поступающих в спортивную секцию, собраны следующие данные: фамилия, имя, год рождения, вес, рост. *Условие поиска*: Дети от 10 до 12 лет, рост которых не менее 155см, а вес не более 45 кг. *Коррекция*: Все дети не младше 5 лет и не старше 14 по году рождения, рост-вес положительные и не более 200.