

## Лабораторная работа №3

### Реализация двусвязных списков с помощью указателей

При программировании использовать линейные **двусвязные** списки.

Элементы списка выбираются в соответствии с вариантом.

**Все элементы обработки данных должны быть оформлены в виде подпрограмм (процедур и/или функций).** Подпрограмма должна возвращать указатель или значение, но не выводить никакой информации, кроме подпрограммы печати списка.

**Не использовать массивы!!!** Осуществлять контроль ошибок (невозможно удалить элемент из пустого списка, добавить элемент к несуществующему списку и т.д.).

**Выполнение лабораторной работы состоит из двух частей: непосредственно написание программ и защита этих программ с полным объяснением каждой строки кода.**

**Лабораторную выполнять в PascalABC или в Lazarus.**

**Программа должна содержать меню (цифровое или любое другое) для выбора операции над списком.**

**Программа должна содержать следующие функции и процедуры:**

- создание пустого списка в соответствии с вариантом;
- проверка списка на пустоту;
- заполнение списка (с помощью генератора случайных чисел и символов) в соответствии с вариантом;
- печать элементов списка в порядке их расположения;
- определение длины списка;
- вставка нового элемента в соответствии с вариантом;
- удаление элемента в соответствии с вариантом;

- разбиение списка на два списка;
- соединение двух линейных списков в один список;
- создание копии списка;
- проверка двух списков на равенство;
- удаление всех элементов списка.
- индивидуальные операции (по вариантам).

### Элементы списка задать в соответствии с вариантом

Вариант	Тип элементов списка	Интервал, которому принадлежат элементы списка	Создание списка	Заполнение списка (добавление элементов)	Вставка нового элемента	Удаление	Индивидуальные операции
1	integer	[-100, 100]	без барьерных элементов	в конец списка	перед $i$ -тым элементом	элемента, следующего за $i$ -тым	Разместить новый элемент перед тем элементом списка, целочисленное поле которого имеет большее значение.
2	string	строки латинских символов длиной 5 символов	с последним барьерным элементом	в конец списка	перед $i$ -тым элементом	элемента, предшествующего $i$ -му	Продублировать в списке все элементы с нечетными номерами (новые элементы добавлять перед су-

							ществующими элементами с такими же значениями)
3	word	[5, 300]	с первым барьерным элементом	в начало списка	непосредственно за i-тым элементом	i –того элемента	Определить количество локальных максимумов в заданном списке. (Локальный максимум в числовом массиве – это последовательность трех рядом стоящих чисел, в которой среднее число больше стоящих слева и справа от него).
4	byte	[0, 80]	с двумя барьерными элементами	в начало списка	непосредственно за i-тым элементом	элемента, следующего за i -тым	Удалить из списка все элементы с нечетными номерами и вывести указатель на первый элемент преобразованного списка. После удаления элементов из списка освободить па-

							мать, которую они занимали
5	char	прописные английские буквы	без барьерных элементов	в конец списка	перед i-тым элементом	i –того элемента	Написать функцию, которая определяет, входит ли список M1 в список M2. Предполагается, что списки существуют.
6	integer	[-250, 250]	с первым барьерным элементом	в начало списка	непосредственно за i-тым элементом	элемента, предшествующего i-му	Определить номер наибольшего из нечетных положительных элементов
7	word	[1000, 2000]	с последним барьерным элементом	в конец списка	перед i-тым элементом	элемента, следующего за i -тым	Даны номера двух различных элементов двусвязного списка. Создать новый список и поместить в него элементы, находящиеся в диапазоне заданных номеров.
8	char	строчные русские буквы	с двумя барьерными	в начало списка	непосредственно за i-тым элементом	i –того элемента	Продублировать в списке все элементы с нечетными

			эlemen- тами				номераи (новые эле- менты добавлять после су- ществующих элементов с такими же значениями)
9	byte	[100,200]	без барь- ерных эле- ментов	в конец списка	перед i-тым элементом	элемента, пред- шествующего i- му	Удалить из списка все эле- менты с нечетными значе- ниями. После удаления элементов из списка осво- бождать память, которую они занимали.
10	real	[-50,30]	с первым барьерным элементом	в начало списка	непосредственно за i- тым элементом	элемента, следу- ющего за i -тым	Переместить в списке дан- ный элемент на K позиций вперед (если после дан- ного элемента находится менее K элементов, то пе- реместить его в конец списка).
11	char	строчные рус- ские буквы от а до н	с послед- ним барь- ерным эле- ментом	в конец списка	перед i-тым элементом	i –того элемента	Написать процедуру, ко- торая вставляет новый элемент перед каждым вхождением заданного

							элемента. Если элементы не найдены, то выдать на экран соответствующие сообщение.
12	word	[40, 100]	с двумя барьерными элементами	в начало списка	непосредственно за i-тым элементом	элемента, предшествующего i-му	Перегруппировать элементы списка, переместив все нечетные элементы в конец списка (в том же порядке).
13	char	цифры	без барьерных элементов	в конец списка	перед i-тым элементом	элемента, следующего за i-тым	Даны два элемента. Создать новый список и поместить в него элементы, значения которых находятся в диапазоне заданных значений.
14	byte	[10, 150]	с первым барьерным элементом	в начало списка	непосредственно за i-тым элементом	i –того элемента	Продублировать в списке все элементы с нечетными значениями (новые элементы добавлять перед существующими элементами).

							тами с такими же значениями)
15	char	знаки препинания, алгебраических операций и др.	с последним барьерным элементом	в конец списка	перед i-тым элементом	элемента, предшествующего i-му	Даны номера двух различных элементов двусвязного списка; элемент с номером X находится в списке перед элементом с номером Y, но не обязательно рядом с ним. Переместить элементы, расположенные между данными элементами (не включая данные элементы), в новый список (в том же порядке). Если новый список окажется пустым, то связанный с ним указатель положить равным nil.
16	integer	[100, 200]	с двумя барьерами	в начало списка	непосредственно за i-тым элементом	элемента, следующего за i-тым	Продублировать в списке все элементы с нечетными

			эlemen- тами				значениями (новые эле- менты добавлять после су- ществующих элементов с такими же значениями)
17	word	[10, 110]	без барь- ерных эле- ментов	в конец списка	перед i-тым элементом	i –того элемента	Разместить новый элемент перед тем элементом списка, целочисленное поле которого имеет наименьшее значение.
18	integer	[-40, 60]	с первым барьерным элементом	в начало списка	непосредственно за i- тым элементом	элемента, пред- шествующего i- му	Переместить в списке дан- ный элемент на K позиций назад (если перед данным элементом находится ме- нее K элементов, то пере- местить его в начало списка).
19	real	[-99, 100]	с двумя ба- рьерными эlemen- тами	в начало списка	непосредственно за i- тым элементом	i –того элемента	Даны номера двух различ- ных элементов двусвяз- ного списка. Создать но- вый список и поместить в



							него элементы, находящиеся в диапазоне заданных номеров. В случае если не было найдено ни одного такого значения, выдается сообщение об ошибке (формат вывода ошибки смотрите выше).
20	byte	[1, 150]	с последним барьерным элементом	в конец списка	перед i-тым элементом	элемента, следующего за i -тым	Исключить из списка все элементы с заданным значением, вычислить их количество. Если элементы не найдены, то выдать на экран соответствующие сообщение.
21	char	строчные английские буквы	без барьерных элементов	в конец списка	перед i-тым элементом	элемента, предшествующего i-му	Даны номера X и Y двух различных элементов двусвязного списка (элемент с номером X находится в списке перед элементом с

							номером Y, но не обязательно рядом с ним). Поменять местами данные элементы.
22	integer	[0, 100]	с первым барьерным элементом	в начало списка	непосредственно за i-тым элементом	элемента, следующего за i -тым	Определить количество локальных минимумов в заданном списке. (Локальный минимум в числовом массиве – это последовательность трех рядом стоящих чисел, в которой среднее число меньше стоящих слева и справа от него).
23	char	цифры и знаки арифметических операций	с последним барьерным элементом	в конец списка	непосредственно за i-тым элементом	i –того элемента	Продублировать в списке первый и последний элементы (новые элементы добавлять после существующих элементов с такими же значениями)

24	char	строчные русские буквы от к до я	с двумя барьерными элементами	в начало списка	перед i-тым элементом	элемента, предшествующего i-му	Даны номера X и Y двух различных элементов двусвязного списка (элемент с номером X находится в списке перед элементом с номером Y, но не обязательно рядом с ним). Поменять местами данные элементы.
25	string	строки длины 6	без барьерных элементов	в конец списка	перед i-тым элементом	элемента, предшествующего i-му	Перегруппировать элементы списка, переместив все строки, начинающиеся с А, Е, О, в конец списка (в том же порядке).
26	string	строки длиной от 2 до 10 символов	с последним барьерным элементом	в конец списка	перед i-тым элементом	элемента, следующего за i-тым	Написать функцию, которая проверяет, упорядочены ли элементы списка по алфавиту.
27	byte	[1, 150]	с последним барьерным	в конец списка	перед i-тым элементом	элемента, следующего за i-тым	Ввести номер элемента. Найти минимальное значение списка из элементов,

			элементом				следующих за заданным элементом, считая его от начала и до конца списка. Если заданное значение отрицательное, считать, что значение номера задается с конца списка.
28	integer	[100, 200]	с двумя барьерными элементами	в начало списка	непосредственно за i-тым элементом	элемента, следующего за i -тым	В заданном списке определить и вывести номера последовательностей чисел, которые монотонно убывают (каждое следующее число меньше предыдущего)
29	real	[-100, 100]	без барьерных элементов	в конец списка	перед i-тым элементом	элемента, следующего за i -тым	Найти минимальный элемент в заданном двусвязном списке
30	word	[5, 300]	с первым барьерным элементом	в начало списка	непосредственно за i-тым элементом	i –того элемента	Найти максимальный элемент в заданном двусвязном списке

31	char	прописные русские буквы	с послед- ним барь- ерным элементом	в конец списка	перед i-тым элементом	элемента, пред- шествующего i- му	Найти количество элемен- тов с заданным значением (ввести с клавиатуры) в за- данном двусвязном списке
32	byte	[0, 80]	с двумя барьер- ными эле- ментами	в начало списка	непосредственно за i- тым элементом	элемента, следу- ющего за i -тым	Найти количество элемен- тов равных нулю в задан- ном двусвязном списке

### Вариант 33. Doubly linked list (Pascal or Lazarus)

(Do not use arrays.) For every operation create procedure or function. Create numeric menu or other.

#### Operations:

- creating an empty list
- testing whether or not a list is empty
- filling the list with random elements
- print the elements of list in the order of occurrence
- find length of a linked list
- inserting a new element as the first one of a list
- insert a new element after a given position
- searching a given element in linked list
- find an element at given position
- find an element with maximum value

- change an element at given position
- deleting the first element of a list
- deleting an element after a given position
- deleting an element with given key
- linked list deletion