

**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ФАКУЛЬТЕТ РАДИОФИЗИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ  
КАФЕДРА ИНФОРМАТИКИ И КОМПЬЮТЕРНЫХ СИСТЕМ**

**Н.В. ЛЕВКОВИЧ  
Н. В. СЕРИКОВА**

**ЗАДАНИЯ ПО КУРСУ**

**«ОСНОВЫ И МЕТОДОЛОГИИ ПРОГРАММИРОВАНИЯ»  
Учебная вычислительная практика**

**ВАРИАНТ В**

**2021  
МИНСК**

## ОГЛАВЛЕНИЕ

<b>1. Функции .....</b>	<b>4</b>
1.1. Передача массивов в функцию .....	4
1.2. Массив слов .....	5
1.3. Возврат ссылок .....	6
1.4. Вычисление корня уравнения. Передача имени функции в качестве параметра. Аргументы по умолчанию .....	8
1.5. Вычисление интеграла. Передача имени функции в качестве параметра .....	10
1.6. Сортировка слиянием .....	11
<b>2. Динамические структуры данных .....</b>	<b>12</b>
2.1. Динамическое выделение памяти для одномерных массивов .....	12
2.2. Динамическое выделение памяти для матриц .....	13
2.3. Линейный список структур .....	14
2.4. Линейные списки как динамическая структура данных .....	15
2.5. Линейные списки как массивы данных .....	15
2.6. Двусвязные списки как динамические структуры данных .....	16
2.7. Бинарное дерево поиска .....	17

## 8 занятий (32 час.)

оценка	количество задач
7	8
8	10
9	12
10	13

№	тема	№ задач			
		7	8	9	10
1	<b>1. Функции</b>	1.1			
2		1.3	1.2		
3		1.4			
4		1.5		1.6	
5	<b>2. Динам. структуры данных</b>	2.1			
6		2.2	2.3		
7		2.4.		2.5	
8		2.6			2.7
9	<b>Зачет</b>				

# 1. ФУНКЦИИ

## 1.1. ПЕРЕДАЧА МАССИВОВ В ФУНКЦИЮ

Выполнить задание, оформив его через функции (ввода, вывода, заполнения). Передачу массива в функции организовать через указатели.

Получить квадратную матрицу порядка  $n$ , элементами которой являются заданные числа  $a_1, \dots, a_{nn}$ , расположенные в ней по спирали (см. схему на рисунке).

Перебор строк/столбцов матрицы осуществить с использованием указателей.

1.

1	2	3	4	5
16	17	18	19	6
15	24	25	20	7
14	23	22	21	8
13	12	11	10	9

2.

9	10	11	12	13
8	21	22	23	14
7	20	25	24	15
6	19	18	17	16
5	4	3	2	1

3.

1	16	15	14	13
2	17	24	23	12
3	18	25	22	11
4	19	20	21	10
5	6	7	8	9

4.

9	8	7	6	5
10	21	20	19	4
11	22	25	18	3
12	23	24	17	2
13	14	15	16	1

5.

5	6	7	8	9
4	19	20	21	10
3	18	25	22	11
2	17	24	23	12
1	16	15	14	13

6.

13	12	11	10	9
14	23	22	21	8
15	24	25	20	7
16	17	18	19	6
1	2	3	4	5

7.

13	14	15	16	1
12	23	24	17	2
11	22	25	18	3
10	21	20	19	4
9	8	7	6	5

8.

5	4	3	2	1
6	19	18	17	16
7	20	25	24	15
8	21	22	23	14
9	10	11	12	13

9.

25	24	23	22	21
10	9	8	7	20
11	2	1	6	19
12	3	4	5	18
13	14	15	16	17

10.

21	22	23	24	25
20	7	8	9	10
19	6	1	2	11
18	5	4	3	12
17	16	15	14	13

11.

25	10	11	12	13
24	9	2	3	14
23	8	1	4	15
22	7	6	5	16
21	20	19	18	17

12.

21	20	19	18	17
22	7	6	5	16
23	8	1	4	15
24	9	2	3	14
25	10	11	12	13

## 1.2. МАССИВ СЛОВ

*Выполнить следующее задание для заданного текста. Оформить задание в виде набора функций: функция создания массива слов, функция получения результирующей строки.*

**Текст** – непустая последовательность символов.

**Слово** – непустая последовательность любых символов, кроме символов-разделителей.

**Предложение** – последовательность слов, разделенных одним или несколькими символами-разделителями.

**Символы-разделители:** «пробел», «.», «,», «:», «;», «!», «?», «-», «(», «)».

1. В заданном тексте найти все пары слов, в которых одно является обращением другого. Записать их в результирующую строку.
2. В заданном тексте найти все слова, которые состоят из одинаковых наборов символов (количество повторений символов не учитывать). Записать их в результирующую строку.
3. В заданном тексте найти все слова, которые имеют одинаковую длину и состоят из одинаковых наборов символов (порядок символов может быть различным). Записать их в результирующую строку.
4. Для каждого слова заданного предложения указать, сколько раз оно встречается в предложении. Записать ответ в результирующую строку.
5. Вывести слова, наиболее часто встречающиеся в строке. Записать их в результирующую строку.
6. Получить текст, в котором слова исходного текста упорядочены по длине слов.
7. Получить текст, в котором слова исходного текста упорядочены по алфавиту.
8. Получить текст, в котором слова исходного текста упорядочены по первой букве каждого слова.
9. Отредактировать заданное предложение, удаляя из него слова, которые уже встречались в предложении раньше.
10. В предложении все слова начинаются с различных букв. Получить текст (если можно), в котором слова предложения расположены в таком порядке, чтобы последняя буква каждого слова совпадала с первой буквой следующего слова.
11. Характеристикой слова назовем длину содержащейся в нем максимальной серии (подряд идущих одинаковых символов). Упорядочить слова заданного предложения в соответствии с ростом их характеристик. Записать их в результирующую строку.
12. Найти множество всех слов (без повторений), которые встречаются в каждом из двух заданных предложений. Записать их в результирующую строку.

### 1.3. ВОЗВРАТ ССЫЛОК

*Напишите функцию, возвращающую ссылку.*

1. Напишите функцию, возвращающую ссылку на минимальное число, встречающееся в заданном массиве произвольного размера (аргумент функции) ровно один раз. Если такого числа нет, то вернуть ссылку на любое из минимальных чисел массива. Замените этот элемент нулевым значением.

2. Напишите функцию, возвращающую ссылку на минимальное число, встречающееся в заданном массиве произвольного размера (аргумент функции) больше одного раза. Если такого числа нет, то вернуть ссылку на минимальное число в массиве. Замените этот элемент значением  $k$ .

3. Напишите функцию, возвращающую ссылку на максимальное число, встречающееся в заданном массиве произвольного размера (аргумент функции) ровно 2 раза. Если такого числа нет, то вернуть ссылку на любое из максимальных чисел массива. Замените этот элемент значением  $k$ .

4. Напишите функцию, возвращающую ссылку на число, встречающееся в заданном массиве произвольного размера (аргумент функции) максимальное количество раз. Если таких чисел несколько, то выбрать минимальное из них. Замените этот элемент значением  $k$ .

5. Напишите функцию, возвращающую ссылку на максимальное число, встречающееся в заданном массиве произвольного размера (аргумент функции) более двух раз. Если такого числа нет, то вернуть ссылку на любое из максимальных чисел массива. Замените этот элемент значением  $k$ .

6. Напишите функцию, возвращающую ссылку на максимальное число, встречающееся в заданном массиве произвольного размера (аргумент функции), являющееся числом Фибоначчи. Если такого числа нет, то вернуть ссылку на любое из максимальных чисел массива. Замените значение этого элемента нулевым значением.

7. Напишите функцию, возвращающую ссылку на максимальное число, встречающееся в заданном массиве произвольного размера (аргумент функции), являющееся степенью числа  $n$ . Если такого числа нет, то вернуть ссылку на любое из максимальных чисел массива. Замените значение этого элемента нулевым значением.

8. Напишите функцию, возвращающую ссылку на максимальное число, встречающееся в заданном массиве произвольного размера (аргумент функции) ровно 1 раз. Если такого числа нет, то вернуть ссылку на любое из максимальных чисел массива. Замените значение этого элемента нулевым значением.

9. Напишите функцию, возвращающую ссылку на максимальное число, встречающееся в заданном массиве произвольного размера (аргумент функции) больше одного раза. Если такого числа нет, то вернуть ссылку на максимальное число массива. Замените значение этого элемента нулевым значением.

**10.** Напишите функцию, возвращающую ссылку на число, встречающееся в заданном массиве произвольного размера (аргумент функции) минимальное количество раз. Если таких чисел несколько, то выбрать максимальное из них. Замените этот элемент значением  $k$ .

**11.** Напишите функцию, возвращающую ссылку на минимальное число, встречающееся в заданном массиве произвольного размера (аргумент функции), являющееся полным квадратом. Если такого числа нет, то вернуть ссылку на любое из максимальных чисел массива. Замените этот элемент значением  $k$ .

**12.** Напишите функцию, возвращающую ссылку на число, встречающееся в заданном массиве произвольного размера (аргумент функции) чаще остальных. Если таких чисел несколько, то выбрать минимальное из них. Замените этот элемент значением  $k$ .

#### 1.4. ВЫЧИСЛЕНИЕ КОРНЯ УРАВНЕНИЯ. ПЕРЕДАЧА ИМЕНИ ФУНКЦИИ В КАЧЕСТВЕ ПАРАМЕТРА. АРГУМЕНТЫ ПО УМОЛЧАНИЮ

Вычислить корень уравнения  $f(x) = 0$  на отрезке  $[a; b]$  с точностью  $\varepsilon = 10^{-6}$ , используя заданный метод ( $M = 1$  – метод половинного деления,  $M = 2$  – метод касательных,  $M = 3$  – метод хорд) для заданных функций. Вычисление корня уравнения оформить в виде функции с функциональным параметром, параметры  $a, b, \varepsilon, s$  – в виде аргументов по умолчанию.

Результат представить в виде таблицы ( $s$  – значение параметра,  $x$  – вычисленный корень уравнения,  $f(x)$  – значение функции в найденной точке  $x$ ,  $k\_iter$  – количество итераций цикла для получения корня с заданной точностью):

$S$	$X$	$F(x)$	$k\_iter$
$S$	...	...	...
$S+\Delta s$	...	...	...
...	...	...	...

- а)  $f(x) = x^2 - 3$   $a = 1; \quad b = 3;$   
 б)  $f(x) = e^{-sx} - 2 + x^2$   $a = 0; \quad b = 1,5; s \in [0,1; 1,3]; \Delta s = 0,3. M=1.$
- а)  $f(x) = x^3 - 3$   $a = 1; \quad b = 4;$   
 б)  $f(x) = \sqrt[5]{x} - 2 \cos^2(\pi x / 2)$   $a = 0; \quad b = 1,5; s \in [0,5; 2], \Delta s = 0,5. M=1.$
- а)  $f(x) = (x-1)^2 - 3$   $a = 1; \quad b = 4;$   
 б)  $f(x) = e^{(x-s)} - \sqrt{x+1}$   $a = 0; \quad b = 2; s \in [0,3; 0,7]; \Delta s = 0,1. M=1.$
- а)  $f(x) = (x-1)^2 - 3$   $a = -2; \quad b = 1;$   
 б)  $f(x) = \cos^2(x) - \sqrt[5]{x}$   $a = 0; \quad b = 1; s \in [1,95; 2], \Delta s = 0,01. M=1.$
- а)  $f(x) = (x-1)^2 - 5$   $a = -3; \quad b = 0;$   
 б)  $f(x) = x^2 - \sin(5x^s)$   $a = 0,5; \quad b = 0,8; s \in [0,7; 1,6], \Delta s = 0,3; M=2.$
- а)  $f(x) = (x-1)^3 - 8$   $a = 1; \quad b = 4;$   
 б)  $f(x) = s \cos^2(\pi x) - \sqrt{x}$   $a = 0; \quad b = 1,5; s \in [0,95; 1,2], \Delta s = 0,05; M=2.$
- а)  $f(x) = (x+3)^3 - 8$   $a = -2; \quad b = 1;$   
 б)  $f(x) = \cos(\pi x) - x^s$   $a = 0; \quad b = 2; s \in [2,8; 3,2], \Delta s = 0,1; M=2.$
- а)  $f(x) = (x-1)^3 - 1$   $a = 0; \quad b = 3;$   
 б)  $f(x) = sx - \cos^2(\pi x)$   $a = -1; \quad b = 0,7; s \in [1; 3], \Delta s = 1; M=1.$
- а)  $f(x) = (x-1)^2 - 5$   $a = 2; \quad b = 4;$   
 б)  $f(x) = (x-s)^2 - e^{-x}$   $a = -1,5; \quad b = 2; s \in [0,7; 1,6], \Delta s = 0,3; M=3.$



- 10.** а)  $f(x) = (x+1)^2 - 5$   $a = 0; \quad b = 2;$   
 б)  $f(x) = x^2 - e^x - 1,5s$   $a = -1,5; \quad b = 1; \quad s \in [0,9; 1,1], \quad \Delta s = 0,05; \quad M=3.$
- 11.** а)  $f(x) = (x+1)^2 - 4$   $a = 0; \quad b = 3;$   
 б)  $f(x) = \cos^2(\pi x) + x^2 - 1,5s$   $a = 0; \quad b = 2; \quad s \in [0,9; 1,2], \quad \Delta s = 0,02; \quad M=3.$
- 12.** а)  $f(x) = (x+1)^2 - 9$   $a = 1; \quad b = 4;$   
 б)  $f(x) = \cos^2(\pi x) - e^{x^s} + 1$   $a = 0; \quad b = 1; \quad s \in [0,96; 1,02], \quad \Delta s = 0,02; \quad M=3.$

## 1.5. ВЫЧИСЛЕНИЕ ИНТЕГРАЛА. ПЕРЕДАЧА ИМЕНИ ФУНКЦИИ В КАЧЕСТВЕ ПАРАМЕТРА

Вычислить следующие интегралы заданным методом ( $M = 1$  – метод левых прямоугольников,  $M = 2$  – метод правых прямоугольников,  $M = 3$  – метод средних прямоугольников,  $M = 4$  – метод трапеции,  $M = 5$  – метод Симпсона.), воспользовавшись критерием двойного пересчета, с точностью  $\varepsilon = 10^{-6}$ .

Значения параметра  $s$ :  $s_i \in [s1; s2]$ ,  $s_i = s1 + i\Delta s$ ,  $i = 0, 1, \dots$ .

Значения параметра  $t$ :  $t_i \in [t1; t2]$ ,  $t_i = t1 + i\Delta t$ ,  $i = 0, 1, \dots$ .

Вычисление значения функции в точке оформить в виде функции, вычисление интеграла – в виде функции с параметром функционального типа для задания функции. Процедуры оформить в виде отдельного файла.

Результаты представить в виде таблицы ( $s, t$  – соответствующие значения параметров,  $Int$  – вычисленное значение интеграла,  $k\_iter$  – количество пересчетов значений интеграла для получения заданной точности):

$s$	$t$	$Int$	$k\_iter$
$s1$	$t1$	...	...
$s1$	$t1+\Delta t$	...	...
...	...	...	...
$s1+\Delta s$	$t1$	...	...
...	...	...	...

1.  $\int_a^b \frac{\arctg(\sqrt[5]{x})}{x^2 + s \sin(x)} dx$

2.  $\int_a^b \sqrt[t]{t + x^3} dx$

3.  $\int_a^b \frac{\sqrt[t]{1+x^2}}{e^{\sin(x)+s}} dx$

4.  $\int_a^b s \frac{\cos^3(x^2)}{\sqrt[t]{x}} dx$

5.  $\int_a^b s \frac{x^2}{\lg(x^t) + \cos^2(\sqrt[3]{x})} dx$

6.  $\int_a^b \frac{1+x^t+x^2}{\sqrt{x^3+1}} (1 - \sin(sx)) dx$

7.  $\int_a^b \frac{\tg(x^2) + \sqrt{x}}{t \lg(x+s)} dx$

8.  $\int_a^b \frac{t}{\sqrt{x^3+1} + s} dx$

9.  $\int_a^b t \frac{3x^s}{\sqrt{1+x^3}} dx$

10.  $\int_a^b \frac{\sin^2(x^t)}{\sqrt{1+x^3}} dx$

11.  $\int_a^b t \frac{1+(sx)^2}{1+x^3} dx$

12.  $\int_a^b t \frac{e^{0,5x}}{\sqrt{x+s}} dx$

**Таблица значений параметров**

№ задания	s1	s2	$\Delta s$	t1	t2	$\Delta t$	a	b	M
1	0	0,5	0,1	0,6	1,8	0,4	0,1	1,2	1
2	1,5	2,5	0,5	0	4	1	0,81	1,762	1
3	0	2	0,5	2	5	1	-1	1	1
4	0	5	1	2	3	0,2	0,1	0,7	2
5	1	10	3	1	3	1	2,63	3,1	2
6	0	0,5	0,1	0,5	2,5	0,5	1	2	3
7	1	4	0,5	0,4	1,2	0,2	0,4	1,2	3
8	0	0,5	0,1	1	3	0,5	1,3	2,621	4
9	1	4	1	0,2	1,2	0,2	0	1,075	4
10	0	1	0,2	1	4	1	s	1,234	5
11	1	5	1	0,5	2,5	0,5	3	4,254	5
12	0,5	2	0,5	0,5	4	0,5	0	1,047	5

### 1.6. СОРТИРОВКА СЛИЯНИЕМ

*Дан массив чисел произвольной длины. Отсортировать массив, используя **сортировку слияниями**.*

*Создать шаблоны и перегрузки функций для сортировки, вывода результатов, заполнения массивов.*

*Выполнить сортировки массивов для данных разных типов: целых, вещественных, символьных, **С-строк**, используя написанные функции.*

## 2. ДИНАМИЧЕСКИЕ СТРУКТУРЫ ДАННЫХ

### 2.1. ДИНАМИЧЕСКОЕ ВЫДЕЛЕНИЕ ПАМЯТИ ДЛЯ ОДНОМЕРНЫХ МАССИВОВ

*Выполнить задание, используя динамическое выделение памяти для одномерного массива. Дана последовательность натуральных чисел, в которой каждое число  $\leq 10^4$ . Последовательность генерируется случайным образом или вводится с клавиатуры. Длина последовательности  $n \leq 10^9$ . Получить итоговый массив за один просмотр элементов (не храня в памяти всю последовательность):*

1. в котором все числа исходной последовательности расположены по возрастанию значений (без повторений);
2. в котором все числа исходной последовательности расположены по убыванию значений (без повторений);
3. образованный из натуральных чисел  $\leq 10^4$ , не встречающихся в исходной последовательности, по возрастанию;
4. образованный из натуральных чисел  $\leq 10^4$ , не встречающихся в исходной последовательности, по убыванию;
5. образованный из исходной последовательности исключением чисел, которые встречаются ровно один раз, по возрастанию значений (без повторений);
6. образованный из чисел, встречающихся в исходной последовательности более одного раза, по возрастанию (без повторений);
7. образованный из чисел, встречающихся в исходной последовательности более одного раза, по убыванию без повторений;
8. образованный из чисел, встречающихся в исходной последовательности ровно два раза, по возрастанию (без повторений);
9. образованный из чисел, встречающихся в исходной последовательности ровно два раза, по убыванию (без повторений);
10. образованный из исходной последовательности исключением повторных вхождений одного и того же числа, элементы должны быть упорядочены по возрастанию значений;
11. в котором все числа исходной последовательности расположены по возрастанию (без повторений);
12. в котором все числа исходной последовательности расположены по убыванию (без повторений).

## 2.2. ДИНАМИЧЕСКОЕ ВЫДЕЛЕНИЕ ПАМЯТИ ДЛЯ МАТРИЦ

*Выполнить задание, используя динамическое выделение памяти для двумерного массива, двумя способами:*

- *описывая двумерный массив как одномерный, с расчетом смещения элемента массива по линейной формуле;*
- *описывая двумерный массив как указатель на массив указателей.*

1. Дана вещественная квадратная матрица  $A$  порядка  $n$ , векторы  $x, y$  с  $n$  элементами. Составить функцию, которая позволяет получить вектор  $c = A \cdot (x + y)$ .
2. Даны вещественные квадратные матрицы  $A, B, C$  порядка  $n$ . Составить функцию для получения матрицы  $D = A \cdot (B + C)$
3. Дана вещественная квадратная матрица  $A$  порядка  $n$ , вектор  $b$  с  $n$  элементами. Составить функцию, которая позволяет получить вектор  $c = A^2 \cdot b$ .
4. Составить функцию, которая позволяет по заданной квадратной матрице размером  $n \times n$  построить вектор длиной  $2n - 1$ , элементы которого – максимумы элементов главной диагонали и диагоналей, параллельных ей.
5. Составить функцию, которая позволяет по заданной квадратной матрице размером  $n \times n$  построить вектор длиной  $2n - 1$ , элементы которого – минимумы элементов побочной диагонали и диагоналей, параллельных ей.
6. Составить функцию, которая позволяет возвести вещественную матрицу произвольного размера в степень  $n$ , где  $n$  – натуральное заданное число.
7. Даны вещественные квадратные матрицы  $A, B$  порядка  $n$ . Составить функцию для получения матрицы  $C = A \cdot B + B \cdot A$ .
8. Дана вещественная квадратная матрица  $A$  порядка  $n$ . Составить функцию для получения матрицы  $B = E + A + A^2 + \dots + A^m$ , где  $m$  – заданное натуральное число.
9. Дана вещественная квадратная матрица  $A$  порядка  $n$ . Составить функцию для получения матрицы  $B = A + A^2 + A^4 + A^8$ .
10. Следом матрицы называется сумма элементов, расположенных на главной диагонали. Даны вещественная квадратная матрица  $A$  порядка  $m$ , натуральное число  $n$ . Составить функцию для получения суммы следов матриц  $A, A^2, \dots, A^n$ .
11. Составить функцию, которая позволяет возводить в квадрат комплексную квадратную матрицу произвольного размера.
12. Составить функцию, которая позволяет возводить в восьмую степень комплексную квадратную матрицу произвольного размера.

### 2.3. ЛИНЕЙНЫЙ СПИСОК СТРУКТУР

*Создать связанную структуру данных – список. Разработать функции для выполнения базовых операций над линейными связанными структурами, содержащими записи со сведениями о студентах: ФИО, дата рождения, курс, успеваемость. При добавлении элемента данных в список обеспечить упорядоченность по алфавиту фамилий. Оценить асимптотическую сложность алгоритма.*

1. По исходному списку определить ФИО самого младшего студента на каждом курсе. Сформировать список из этих студентов, удалив их из исходного списка.
2. По исходному списку определить ФИО самого старшего студента на каждом курсе. Сформировать список из этих студентов, удалив их из исходного списка.
3. По исходному списку определить всех студентов, ФИО которых начинается на заданную букву. Сформировать список из этих студентов, удалив их из исходного списка.
4. По исходному списку определить всех студентов, возраст которых превышает заданный. Сформировать список из этих студентов, удалив их из исходного списка.
5. По исходному списку определить всех студентов  $n$  курса. Сформировать список из этих студентов, удалив их из исходного списка.
6. По исходному списку определить всех отличников. Сформировать список из этих студентов, удалив их из исходного списка.
7. По исходному списку определить всех отличников  $n$  курса. Сформировать список из этих студентов, удалив их из исходного списка.
8. По исходному списку определить всех неуспевающих студентов. Сформировать список из этих студентов, удалив их из исходного списка.
9. По исходному списку определить всех неуспевающих студентов  $n$  курса. Сформировать список из этих студентов, удалив их из исходного списка.
10. По исходному списку определить студентов, имеющих средний бал успеваемости выше общего среднего бала. Сформировать список из этих студентов, удалив их из исходного списка.
11. По исходному списку определить студентов на  $n$  курсе, имеющих средний бал успеваемости выше среднего бала по его курсу. Сформировать список из этих студентов, удалив их из исходного списка.
12. По исходному списку определить студентов на  $n$  курсе, имеющих средний бал успеваемости выше среднего бала по его курсу. Сформировать список из этих студентов, удалив их из исходного списка.

## 2.4. ЛИНЕЙНЫЕ СПИСКИ КАК ДИНАМИЧЕСКАЯ СТРУКТУРА ДАННЫХ

*Выполнить задания, используя связанные динамические структуры данных «список» в виде связных компонент. Оценить асимптотическую сложность алгоритма.*

1. Описать функцию, которая формирует список символов  $L_2$ , включив в него те символы из  $L_1$ , которые входят в него по одному разу.
2. Описать функцию, которая формирует список символов  $L_2$ , включив в него те символы из  $L_1$ , которые входят в него более одного раза.
3. Описать функцию, которая формирует список символов  $L_2$ , включив в него те символы из  $L_1$ , которые входят в него ровно два раза.
4. Описать функцию, которая формирует список символов-букв  $L_2$ , включив в него те символы-буквы, которые не входят в список  $L_1$ .
5. Описать функцию, которая формирует список символов  $L$ , включив в него по одному разу элементы, которые входят хотя бы в один из списков  $L_1$  и  $L_2$ .
6. Описать функцию, которая формирует список символов  $L$ , включив в него по одному разу элементы, которые входят одновременно в оба списка  $L_1$  и  $L_2$ .
7. Описать функцию, которая формирует список символов  $L$ , включив в него по одному разу элементы, которые входят в один из списков  $L_1$  и  $L_2$ , но в то же время не входят в другой из них.
8. Описать функцию, которая формирует список символов  $L$ , включив в него по одному разу элементы, которые входят в список  $L_1$  и не входят в список  $L_2$ .
9. Описать функцию, которая из списка символов  $L$  удаляет все вхождения списка  $L_1$  (если такой есть).
10. Описать функцию, которая в списке символов  $L$  заменяет первое вхождение списка  $L_1$  (если такой есть) на список  $L_2$ .
11. Описать функцию, которая в списке символов  $L$  заменяет все вхождения списка  $L_1$  (если такой есть) на список  $L_2$ .
12. Описать функцию, которая в списке символов  $L$  заменяет последнее вхождение списка  $L_1$  (если такой есть) на список  $L_2$ .

## 2.5. ЛИНЕЙНЫЕ СПИСКИ КАК МАССИВЫ ДАННЫХ

*Выполнить задания 2.4, используя связанную динамическую структуру данных «список» в виде массивов. Оценить асимптотическую сложность алгоритма.*

## 2.6. ДВУСВЯЗНЫЕ СПИСКИ КАК ДИНАМИЧЕСКИЕ СТРУКТУРЫ ДАННЫХ

*Выполнить задания, используя двусвязные динамические структуры данных в виде связанных компонент. Оценить асимптотическую сложность алгоритма.*

1. Даны натуральное число  $n$ , действительные числа  $x_1, x_2, \dots, x_n$ . Разработать программу вычисления значения выражения следующего вида:  
$$x_1 \cdot x_n + x_2 \cdot x_{n-1} + \dots + x_n x_1.$$
2. Даны натуральное число  $n$ , действительные числа  $x_1, x_2, \dots, x_n$ . Разработать программу вычисления значения выражения следующего вида:  
$$(x_1 + x_n) \cdot (x_2 + x_{n-1}) \cdot \dots \cdot (x_n + x_1).$$
3. Даны натуральное число  $n$ , действительные числа  $x_1, x_2, \dots, x_n$ . Разработать программу вычисления значения выражения следующего вида:  
$$(x_1 + x_2 + nx_n) + (x_2 + x_3 + (n-1)x_{n-1}) + \dots + (x_{n-1} + x_n + 2x_2).$$
4. Даны натуральное число  $n$ , действительные числа  $x_1, x_2, \dots, x_n$ . Разработать программу вычисления значения выражения следующего вида:  
$$(x_1 + x_2 + 2x_n) + (x_2 + x_3 + 2x_{n-1}) + \dots + (x_{n-1} + x_n + 2x_2).$$
5. Даны натуральное число  $n$ , действительные числа  $a_1, a_2, \dots, a_{2n}$ . Получить  
$$(a_1 - a_{2n}) \cdot (a_3 - a_{2n-2}) \cdot (a_5 - a_{2n-4}) \cdot \dots \cdot (a_{2n-1} - a_2).$$
6. Даны натуральное число  $n$ , действительные числа  $a_1, a_2, \dots, a_{2n}$ . Получить  
$$(a_n - a_{n+1}) \cdot (a_{n-1} - a_{n+2}) \cdot (a_{n-2} - a_{n+3}) \cdot \dots \cdot (a_1 - a_{2n}).$$
7. Даны натуральное число  $n$ , действительные числа  $a_1, a_2, \dots, a_{2n}$ . Получить  
$$a_1 a_{2n} + a_2 a_{2n-1} + \dots + a_n a_{n+1}.$$
8. Даны натуральное число  $n$ , действительные числа  $a_1, a_2, \dots, a_{2n}$ . Получить  
$$a_n a_{n+1} + a_{n-1} a_{n+2} + \dots + a_1 a_{2n}.$$
9. Даны натуральное число  $n$ , действительные числа  $a_1, a_2, \dots, a_{2n}$ . Получить  
$$\min(a_1 + a_{2n}, a_2 + a_{2n-1}, \dots, a_n + a_{n+1}).$$
10. Даны натуральное число  $n$ , действительные числа  $a_1, a_2, \dots, a_{2n}$ . Получить  
$$\max(a_1 + a_{2n}, a_2 + a_{2n-1}, \dots, a_n + a_{n+1}).$$
11. Даны натуральное число  $n$ , действительные числа  $a_1, a_2, \dots, a_{2n}$ . Получить  
$$\max(\min(a_1, a_{2n}), \min(a_2, a_{2n-1}), \dots, \min(a_n, a_{n+1})).$$
12. Даны натуральное число  $n$ , действительные числа  $a_1, a_2, \dots, a_{2n}$ . Получить  
$$\min(\max(a_1, a_{2n}), \max(a_2, a_{2n-1}), \dots, \max(a_n, a_{n+1})).$$



## 2.7. БИНАРНОЕ ДЕРЕВО ПОИСКА

*По заданной последовательности различных целых чисел построить соответствующее бинарное дерево поиска  $T$  как динамическую структуру данных. Выполнить следующие задания и вывести элементы дерева на экран. Оценить асимптотическую сложность алгоритма.*

1. Определяет значение самого левого листа дерева.
2. Определить число листьев дерева.
3. Удалить вершину с минимальным значением элемента.
4. Удалить вершину с максимальным значением элемента.
5. Определить число элементов  $k$ -ого уровня.
6. Вывести на экран все листья дерева.
7. Определить номер уровня, в котором содержится максимальное количество вершин.
8. Определить максимальную глубину дерева.
9. Удалить все листья дерева.
10. Определить число ветвей от корня до вершины с заданным элементом. Вывести часть дерева от вершины до данного элемента на экран.
11. Определить число ветвей  $n$ -го уровня этого дерева.
12. Дополнить полученное дерево новыми вершинами так, чтобы каждая вершина, которая не является листом, имела ровно двух преемников, значения элементов в дополнительных вершинах задавать равными значениям их предшественников.