

ЛАБОРАТОРНАЯ РАБОТА №2

«ЛИНЕЙНЫЕ И ЦИКЛИЧЕСКИЕ СПИСКИ»

1.1 Цель работы

Целью работы является изучение структур данных «линейный список» и «циклический список», а также получение практических навыков их реализации.

1.2 Цель работы

Целью работы является изучение структур данных «линейный список» и «циклический список», а также получение практических навыков их реализации.

1.3 Задание на лабораторную работу

Реализовать структуры данных «линейный список» и «циклический список» в соответствии с заданным вариантом. Дополнительно программа должна осуществлять следующие операции:

- 1) Добавление/удаление элемента в список (с клавиатуры);
- 2) Вывод исходного и результирующего списков на экран;
- 3) Если списки являются многочленами, в выводе должна быть отражена степень каждого элемента.

Варианты задания приведены в таблице 2. Элементы последовательности, или коэффициенты многочлена (в зависимости от варианта) – числовые значения элемента списка, количество таких элементов списка равно длине последовательности, или количеству коэффициентов многочлена.

Таблица 1

№ ва р.	Задача	Вид списка
1	Дана последовательность повторяющихся целых чисел a_1, a_2, \dots, a_n . Получить последовательность k_1, k_2, \dots, k_m , содержащую повторяющиеся в исходной последовательности элементы (c) в порядке убывания частоты их повторения в исходной последовательности (count). Неповторяющиеся элементы не включать. $k_1 = \max(\text{count}(c)), k_2 = \min(\max(\text{count}(c), k_1)) \dots$	Линейный односвязный
2	Дана последовательность неповторяющихся чисел a_1, a_2, \dots, a_m . Получить две последовательности. Первая – список целых частей каждого числа в исходной последовательности. Вторая – список дробных частей, если элемент исходной последовательности является целым числом, такой элемент не включается в результирующие последовательности. $B = \text{whole}(a_1), \text{whole}(a_2) \dots \quad K = \text{fract}(a_1), \text{fract}(a_2) \dots$	Линейный односвязный
3	Даны натуральное число n и символы s_1, s_2, \dots, s_n . Получить символы, принадлежащие последовательности s_1, s_2, \dots, s_n , которые входят в нее по одному разу	Линейный односвязный
4	Даны 2 многочлена. Каждый многочлен $P(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$ с целыми коэффициентами можно представить в виде списка, причем если $a_i = 0$, то соответствующее звено не включать в список. Определить процедуру, которая строит многочлен p – сумму многочленов q и r	Линейный односвязный
5	Дано натуральное число n и целые числа a_1, a_2, \dots, a_n . Вычислить $\min_{1 \leq i \leq n} a_i - \bar{a} $, где \bar{a} среднее арифметическое чисел a_1, a_2, \dots, a_n	Линейный односвязный
6	Дано натуральное число n и целые числа a_1, a_2, \dots, a_n . Требуется получить последовательность $x_1, x_2, \dots, x_k, y_1, y_2, \dots, y_k$, где x_1, \dots, x_k – взятые в порядке следования (слева на право) четные члены последовательности a_1, \dots, a_n , а y_1, \dots, y_k – нечетные члены	Линейный односвязный
7	Дана последовательность неповторяющихся целых чисел a_1, a_2, \dots, a_n . Получить последовательность, содержащую наименьший элемент исходной последовательности, и все предшествующие ему. $K = \min(a_1, a_2, \dots, a_n), a_{\min-1}, a_{\min-2}, \dots, a_1$	Линейный двусвязный
8	Дана последовательность неповторяющихся целых чисел a_1, a_2, \dots, a_n , где $n > 4$. Получить последовательность, содержащую элементы исходной последовательности с удаленными двумя наименьшими и двумя наибольшими элементами. $K = A - (\max(a), \min(a), \max(a - \max), \min(a - \min))$	Линейный двусвязный
9	Дана последовательность латинских букв, оканчивающаяся точкой. Среди букв есть специальный символ, появление которого означает отмену предыдущей буквы; n знаков подряд отменяют n предыдущих букв, если такие есть. Учитывая вхождение этого символа преобразовать последовательность.	Линейный двусвязный

№ ва р.	Задача	Вид списка
10	Даны натуральные числа k, m, n и последовательности символов $s_1, s_2, \dots, s_k, t_1, t_2, \dots, t_m, u_1, u_2, \dots, u_n$. Получить по одному разу те символы, которые входят во все три последовательности, но расположить их по возрастанию.	Линейный двусвязный
11	Дана последовательность неповторяющихся чисел a_1, a_2, \dots, a_n и некое число c , принадлежащее данной последовательности. Составить 2 последовательности. Первая – все числа, находящиеся до указанного числа в обратном порядке. Вторая – все числа после указанного числа в прямом порядке.	Линейный двусвязный
12	Дана последовательность неповторяющихся чисел a_1, a_2, \dots, a_n . Получить последовательность, содержащую все числа в заданном диапазоне $[f..k]$, которые встречаются в исходной последовательности.	Линейный двусвязный
13	Дана последовательность повторяющихся целых чисел a_1, a_2, \dots, a_n . Получить последовательность, содержащую среднее арифметическое n элементов исходной последовательности, затем $n-1$ элементов и т.д. $K = \text{average}(a_1 \dots a_n), \text{average}(a_1 \dots a_{n-1}), \dots, \text{average}(a_1, a_2), a_1$	Циклический односвязный
14	Дана последовательность неповторяющихся чисел a_1, a_2, \dots, a_n . Получить 3 последовательности. Первая содержит элементы исходной последовательности, делящиеся без остатка на 3. Вторая – элементы, делящиеся без остатка на 2. Третья последовательность содержит элементы первой и второй полученных последовательностей, за исключением элементов, которые в них дублируются. Пример: $A=1,3,6,7,4,2$; $K=3,6$; $B=6,4,2$; $\text{Res}=3,4,2$	Циклический односвязный
15	Дана последовательность неповторяющихся целых чисел a_1, a_2, \dots, a_n . Получить последовательность, содержащую $n/2$ элементов. Первый элемент такой последовательности – минимальный элемент исходной последовательности b_1 , второй ближайшее число к $b_1 * 2$, третий – ближайшее к $b_2 * 3$ и т.д. $b_1 = \min(a), \min(a_1 - b_1 * 2, a_2 - b_1 * 2, \dots, a_n - b_1 * 2), \dots$	Циклический односвязный
16	Многочлен $P(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$ с целыми коэффициентами можно представить в виде списка, причем если $a_i = 0$, то соответствующее звено не включать в список. Определить процедуру, которая строит многочлен q – производная многочлена p .	Циклический односвязный
17	Многочлен $P(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$ с целыми коэффициентами можно представить в виде списка, причем если $a_i = 0$, то соответствующее звено не включать в список. Определить логическую функцию $\text{Equal}(p, q)$, проверяющие на равенство многочлены p и q	Циклический односвязный
18	Проверить, удовлетворяют ли элементы списка (базовый тип integer) закону $x = f(x_0, h)$, где x – элемент списка, h – шаг, x_0 – начальный элемент списка. Пример: $x_0=5, h=1, x_1=6, x_2=7, x_3=8 \dots$ Элементы списка удовлетворяют закону $x=h(5,1)$	Циклический односвязный

№ ва р.	Задача	Вид списка
19	Дана последовательность неповторяющихся целых чисел a_1, a_2, \dots, a_n . Получить последовательность, содержащую поочередно каждый четный элемент исходной последовательности слева – направо, и каждый нечетный элемент последовательности справа – налево. Если n -четное, то: $a_1, a_m, a_2, a_{n-1}, \dots$	Циклический двусвязный
20	Дана последовательность чисел a_1, a_2, \dots, a_n и число K . Необходимо суммировать элементы исходной последовательности, пока сумма не будет больше или равна K , после чего выводятся все просуммированные элементы.	Циклический двусвязный
21	Дана последовательность неповторяющихся целых чисел a_1, a_2, \dots, a_n . Получить последовательность длиной, содержащую разность между первым и последним элементом, вторым и предпоследним и т.д. Если n – нечетное, центральный элемент копируется без изменений.	Циклический двусвязный
22	Дана последовательность неповторяющихся чисел a_1, a_2, \dots, a_n . Получить 2 последовательности, первая из которых является зеркальным отображением исходной. Вторая – поэлементная разница между исходной и второй последовательностью.	Циклический двусвязный
23	Дана последовательность неповторяющихся чисел a_1, a_2, \dots, a_n . Получить 3 последовательности. Первая – копия исходной последовательности с инвертированными знаками. Вторая – копия исходной последовательности со взятыми по модулю значениями. Третья – поэлементная сумма первой и второй последовательностей, с удаленными из неё нулевыми значениями.	Циклический двусвязный

1.4 Порядок выполнения работы

- 1) выбрать вариант задания из подраздела 1.3 в соответствии с требованиями;
- 2) изучить теоретический материал, изложенный в учебном пособии;
- 3) разработать на языке программирования высокого уровня программу, выполняющую поставленную задачу с использованием заданной структуры данных;
- 4) написать отчет о работе;
- 5) защитить отчет.

К защите отчета по лабораторной работе, включающую демонстрацию работы программы, необходимо сформировать два или более контрольных примера.

1.5 Содержание отчета

Отчет должен содержать:

- 1) титульный лист;
- 2) цель работы;
- 3) вариант задания;
- 4) листинг программы, реализующей поставленную задачу с использованием заданных структур данных;
- 5) контрольные примеры;
- 6) выводы по работе.

1.6 Пример выполнения работы

Предположим, что необходимо выполнить следующий вариант задания:

№ вар.	Задача	Вид списка
24	Дано натуральное число n и целые числа a_1, a_2, \dots, a_n . Требуется получить последовательность $x_1, x_2, \dots, x_k, y_1, y_2, \dots, y_k$, где x_1, \dots, x_k – взятые в порядке следования (слева на право) четные члены последовательности a_1, \dots, a_n , а y_1, \dots, y_k – нечетные члены	Линейный

Для выполнения поставленного задания потребуется организовать два списка:

- линейный список A , состоящий из n элементов и содержащих исходные целые числа;
- линейный список X , состоящий из n элементов и содержащих в первых k элементах числа x_1, x_2, \dots, x_k , а в последующих элементах – числа y_1, y_2, \dots, y_k .

Элементы обоих списков описываются одинаково.

Теперь можно разработать программу, которая должна выполнять следующие шаги:

Содержимое структуры:

```
struct LinearList
{
    int coeff;
```

```

    LinearList *next;
};

```

1) создание и заполнение линейного списка A:

```

void AddListElem(int NewListElem, LinearList *&First)
{
    LinearList *NewElem = new LinearList;
    NewElem->next = NULL;
    NewElem->coeff = NewListElem;
    if (First == NULL)
        First = NewElem;
    else
    {
        LinearList *tmp = First;
        while (tmp->next != NULL)
        {
            tmp = tmp->next;
        }
        tmp->next = NewElem;
    }
}

```

2) создание и заполнение линейного списка X;

Perv – указатель на первый элемент списка.

FinalSequence – указатель на первый элемент результирующего списка

```

void GetSequence(LinearList *Perv, LinearList *&FinalSequence)
{
    LinearList *ResultCarriage = new LinearList;
    LinearList *Carriage = new LinearList;
    Carriage = Perv;
    //Участок с обходом четных элементов
    if (Carriage->next != NULL)
    {
        Carriage = Carriage->next;
        FinalSequence = new LinearList;
        FinalSequence->coeff = Carriage->coeff;
        ResultCarriage = FinalSequence;
        while (Carriage->next != NULL)
        {
            if (Carriage->next->next != NULL)
            {
                Carriage = Carriage->next->next;
                ResultCarriage->next = new LinearList;
                ResultCarriage->next->coeff = Carriage->coeff;
                ResultCarriage = ResultCarriage->next;
            }
            else break;
        }
    }
    else
    {
        FinalSequence = new LinearList;
    }
}

```

```

    FinalSequence->coeff = Carriage->coeff;
    return;
}
//Участок с обходом нечетных элементов
Carriage = Perv;
ResultCarriage->next = new LinearList;
ResultCarriage->next->coeff=Carriage->coeff;
ResultCarriage = ResultCarriage->next;
while (Carriage->next != NULL)
{
    if (Carriage->next->next != NULL)
    {
        Carriage = Carriage->next->next;
        ResultCarriage->next = new LinearList;
        ResultCarriage->next->coeff = Carriage->coeff;
        ResultCarriage = ResultCarriage->next;
    }
    else break;
}
ResultCarriage->next = NULL;
}

```

Для выполнения поставленной задачи совершается последовательный обход исходного линейного списка сначала по чётным элементам, затем по нечётным. При этом необходимо проверять наличие элемента в участке памяти, в который происходит обращение программы при очередной итерации. При значении NULL происходит выход из цикла. Такой подход позволит сохранить исходный список, а так же не выполнять лишние итерации при обходе.

1.7 Контрольные вопросы

- 1) Что такое список?
- 2) Назовите основные характеристики линейного однонаправленного списка.
- 3) На какой элемент линейного однонаправленного списка необходимо обеспечивать позиционирование какого-либо указателя? Обоснуйте ответ.
- 4) Назовите основные характеристики линейного двунаправленного списка.
- 5) Сравните линейный однонаправленный список и линейный двунаправленный список.
- 6) Назовите основные характеристики циклического однонаправленного списка.

- 7) Сравните линейный однонаправленный список и циклический однонаправленный список.
- 8) Назовите основные характеристики циклического двунаправленного списка.
- 9) Сравните линейный двунаправленный список и циклический двунаправленный список.