# Лабораторная работа №1

## Варианты заданий для лабораторной работы № 1 Часть 1

Вариант № 1

Движение без топлива? Владелец автомобиля приобрел новый карбюратор, который экономит 50 % топлива, новую систему зажигания, которая экономит 30 % топлива, и поршневые кольца, экономящие 20 % топлива. Верно ли, что его автомобиль теперь сможет обходиться совсем без топлива? Найти фактическую экономию для произвольно заданных сэкономленных процентов.

Вариант № 2

Задача жестянщика. Из круга радиуса r вырезан прямоугольник, большая сторона которого равна a. Найти максимальный радиус круга, который можно вырезать из этого прямоугольника?

Вариант № 3

Приближение sin x. Функция y= sin x на отрезке  хорошо аппроксимируется разложением: . Для заданного значения аргумента x вычислить y по этой формуле и сравнить с точным значением, вычисленным с помощью стандартной функции *sin*.

Вариант № 4

Переправа. Пловцу надо под прямым углом к фарватеру преодолеть реку шириной b м. Его скорость в стоячей воде ; скорость течения реки - . Под каким углом к фарватеру он должен плыть, чтобы его «не снесло»? Сколько времени займет переправа? Как изменится решение, если посередине реки скорость пловца упадет с  до ?

Вариант № 5

Вершина параболы. Найти координаты вершины параболы 

Вариант № 6

Треугольник задается координатами своих вершин на плоскости: , , . Найти площадь треугольника ABC.

Вариант № 7

Угол α задан в градусах, минутах и секундах. Найти его величину в радианах (с максимально возможной точностью).

Вариант № 8

Угол α задан в радианах. Найти его величину в градусах, минутах и секундах.

Вариант № 9

Длина отрезка задана в дюймах (1 дюйм=2,54 см). Перевести значение длины в метрическую систему, т.е. выразить ее в метрах, сантиметрах и миллиметрах. Так, например, 21 дюйм = 0 м 53 см 3,4 мм.

Вариант № 10

Заданы моменты начала и конца некоторого промежутка времени в часах, минутах и секундах (в пределах одних суток). Найти продолжительность этого промежутка в тех же единицах измерения.

Вариант № 11

Кубическое уравнение. Заданы три корня кубического уравнения: x1, x2, x3. Найти коэффициенты этого уравнения.

Вариант № 12

Квадратное уравнение. Найти корни квадратного уравнения, заданного своими коэффициентами, с положительным дискриминантом. Подстановкой в уравнение убедиться в погрешности вычислений.

Вариант № 13

Комплексное число. Заданы действительная и мнимая части комплексного числа *z=x+ iy*. Преобразовать его в тригонометрическую форму и напечатать в виде выражения: *z=r(cosφ+i sinφ)*.

Для справки:

; .

Вариант № 14

Треугольник задается координатами своих вершин на плоскости: , , . Найти сумму длин медиан треугольника ABC.

Вариант № 15

Треугольник задается координатами своих вершин на плоскости: , , . Найти точку пересечения биссектрис треугольника ABC (центр вписанной с него окружности).

Вариант № 16

Треугольник задается координатами своих вершин на плоскости: , , . Найти внутренние углы треугольника ABC (в градусах).

Вариант № 17

Треугольник задается координатами своих вершин на плоскости: , , . Найти длину и основание высоты, опущенной из вершины A на сторону BC.

Вариант № 18

Треугольник задается координатами своих вершин на плоскости: , , . Найти точку D, симметричную точке A относительно стороны BC.

Вариант № 19

В равнобедренном прямоугольном треугольнике известна высота h, опущенная на гипотенузу. Найти стороны треугольника.

Вариант № 20

«Косой» квадрат. У квадрата ABCD на плоскости известны координаты двух противоположных вершин – точек A и C. Найти координаты точек B и D.

Примечание. Расположение квадрата произвольно, его стороны не обязательно параллельны координатным осям.

Вариант № 21

Треугольник ABC задан длинами своих сторон. Найти длину высоты, опущенной из вершины A.

Вариант № 22

Заданы уравнения двух пересекающихся прямых на плоскости: , . Найти (в градусах, минутах и секундах) угол между ними, используя формулу: .

Вариант № 23

Трехмерные вектора заданы своими координатами, например, , . Найти угол в градусах между векторами A и B, используя формулу: .

Вариант № 24

Трехмерные вектора заданы своими координатами, например, , , . Найти объем пирамиды, построенной на векторах A, B, C, как на сторонах.

Вариант № 25

Трехмерные вектора заданы своими координатами, например, , , . Найти длину диагонали параллелепипеда, построенного на векторах A, B, C, как на сторонах.

Вариант № 26

Определить периметр правильного n-угольника, описанного около окружности радиуса r.

Вариант № 27

Известна длина окружности. Найти площадь круга, ограниченного этой окружностью.

## Варианты заданий для лабораторной работы № 1 Часть 2

Вариант № 1

Кирпич. Пройдет ли кирпич со сторонами a, b и c через прямоугольное отверстие со сторонами r и s? Стороны отверстия должны быть параллельны граням кирпича.

Вариант № 2

Шар и ромб. Может ли шар радиуса r пройти через ромбообразное отверстие с диагоналями p и q?

Вариант № 3

Вычислите значение выражения y= sqrt(2Х-1) в случае, когда это возможно (т.е. если (2Х-1)>=0). В противном случае выводите на сообщение «Задайте другое значение Х»

Вариант № 4

Запрашивайте рост и вес пользователя вашей программы. В случае, если вес больше, чем рост -100, то выводите сообщение «Хорошо бы Вам похудеть». В противном случае выводите сообщение «Ваш вес в норме».

Вариант № 5

Если целое число М делится нацело на целое число N, то вывести на экран частное от деления, в противном случае вывести сообщение М на N нацело не делится.

Вариант № 6

Запишите условный оператор, в котором значение переменной с вычисляется по формуле a+b, если а - нечетное и a\*b, если а - четное.

Вариант № 7

Написать программу для подсчета суммы только положительных из трех данных чисел.

Вариант № 8

Заданы стороны треугольника: a, b, c. Проверить, является ли этот треугольник остроугольным.

Вариант № 9

Заданы стороны треугольника: a, b, c. Проверить, является ли этот треугольник тупоугольным.

Вариант № 10

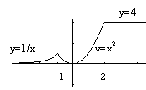
Заданы три параметра: a, b, c. Проверить, могут ли эти параметры быть длинами сторон треугольника.

Вариант № 11

Даны целые положительные числа x, y, z. Выяснить, существует ли треугольник с углами, равными x, y и z, если x, y и z представляют собой градусы.

Вариант № 12

Дано действительное а. Для функции f(a), график которой представлен на рисунке, вычислить f(a).



|  |  |
| --- | --- |
|  |  |

Вариант № 13

Даны три действительных числа. Выбрать из них те, которые принадлежат интервалу (1, 3). Или вывести сообщение, что таких чисел нет.

Вариант № 14

Даны действительные числа x, y. Если x, y отрицательны, то каждое значение заменить его модулем; если отрицательное только одно из них, то оба значения увеличить на 0.5; если оба значения не отрицательны и ни одно из них не принадлежит отрезку [0.5, 2.0], то оба значения уменьшить в 10 раз; в остальных случаях x, y оставить без изменения.

Вариант № 15

Определить и вывести на печать номер квадранта, в котором расположена точка М(x,y), x и y заданные вещественные числа.

Вариант № 16

Из величин, определяемых выражениями a=sinx, b=cosx, c=ln|x| при заданном х, определить и вывести на экран дисплея минимальное значение.

Вариант № 17

Определить, какая из двух точек - M1(x1, y1) или M2(x2, y2) - расположена ближе к началу координат. Вывести на экран дисплея координаты этой точки.

Вариант № 18

Определить, какая из двух фигур (круг или квадрат) имеет большую площадь. Известно, что сторона квадрата равна а, радиус круга r. Вывести на экран название и значение площади большей фигуры.

Вариант № 19

Определить, попадает ли точка M(x, y) в круг радиусом r с центром в точке (x0, y0)

Вариант № 20

Найти количество положительных (отрицательных) чисел среди четырех целых чисел A, B, C, D.

Вариант № 21

Написать программу, определяющую, есть ли в введенном числе дробная часть. Написать программу, определяющую, есть ли в введенном числе дробная часть.

Вариант № 22

Даны три значения. Вывести большее из них.

Вариант № 23

Даны три переменные вещественного типа: A, B, C. Если их значения упорядочены по возрастанию, то удвоить их; в противном случае заменить значение каждой переменной на значение с противоположным знаком. Вывести новые значения переменных A, B, C.

Вариант № 24

Даны вещественные числа a, b, c (a ≠ 0). Выяснить, имеет ли уравнение ах2 + bx + c = 0 вещественные корни.

Вариант № 25

Известны две скорости: одна в километрах в час, другая — в метрах в секунду. Какая из скоростей больше?

Вариант № 26

Даны радиус круга и сторона квадрата, y какой фигуры площадь больше?

Вариант № 27

Даны объемы и массы двух тел из разных материалов. Материал какого из тел имеет большую плотность?

Вариант № 28

Известны сопротивления двух не соединенных друг с другом участков электрической цепи и напряжение на каждом из них. По какому участку протекает меньший ток?

## Что нужно знать в результате выполнения работы

1. Операторы ввода/вывода в С++.
2. Типы данных в С++ (сам факт статической типизации).
3. Операторы ветвления If и Switch.
4. Полная форма оператора If (If, If else, else).
5. Полная форма оператора Switch (несколько case + default).
6. Уметь писать программу с несколькими вложенными If.
7. Понимать, что означают спецификаторы signed и unsigned, зачем они применяются.
8. Понимать, что такое целочисленное переполнение, что это такое.
9. Преобразование типов в С++ (явное и неявное).
10. Понимать, что означает ключевое слово auto.

# Лабораторная работа №2

## Краткая теоретическая справка

Операторы цикла используются для организации многократно повторяющихся вычислений. Тело цикла – оператор или составной оператор (блок) может выполняться несколько раз, однократное выполнение тела цикла называется итерацией. При переходе от итерации к итерации некоторые переменные могут менять свои значения.

В языке Си существует три разновидности операторов цикла.

1. Цикл с предусловием имеет формат:

*while(<выражение\_условие>) <оператор>*

*<выражение\_условие>* - значение этого выражения может быть любого типа языка Си, заменяющего логический (целый, вещественный, указатель), значение условия считается истинным, если оно отлично от нуля, при этом выполняется тело цикла, телом цикла может быть составной оператор или блок.

1. Цикл с постусловием имеет формат:

*do <оператор> while(<выражение\_условие>);*

Цикл работает аналогично предыдущему, но только в начале выполняется оператор – тело цикла, затем проверяется выражение – условие.

1. Цикл *for* имеет формат:

*for([<инициализаторы>];[<выражение\_условие>]; [<модификации>]) <оператор>*

*<инициализаторы>* используются для объявления и присвоения начальных значений переменным, используемым в цикле, можно записать несколько выражений, разделенных запятой.

*<выражение\_условие>* определяет условия продолжения цикла, задается аналогично, как в предыдущих видах циклов.

*<модификации>* выполняются после каждой итерации цикла и служат обычно для изменения параметров цикла, здесь можно записать несколько выражений через запятую.

## Варианты заданий для лабораторной работы № 2

Вариант № 1

Проверить численно первый замечательный предел , задавая значения x: 1; ½; ¼; 1/8; … до тех пор, пока левая часть равенства не будет отличаться от правой менее чем на заданную погрешность ξ=10-2,…, 10-6, определить как меняется число итераций в зависимости от ξ.

Вариант № 2

Проверить численно второй замечательный предел , задавая значения n: 1; 2; 3; 4; … При каком *n* исследуемое выражение отличается от *e* менее чем на заданную погрешность ξ=10-2,…, 10-6?

Вариант № 3

Сравнить скорость сходимости (число слагаемых для достижения заданной точности ξ=10-2,…, 10-6) следующих разложений числа π:

;

.

Вариант № 4

Найти сумму первых k чисел последовательности Фибоначчи. Последовательность определяется законом:

F0=F1=1; Fn=Fn-1+Fn-2 - для n≥2.

Вариант № 5

Определить первое число последовательности Фибоначчи, превышающее число А. Последовательность определяется законом:

F0=F1=1; Fn=Fn-1+Fn-2 - для n≥2.

Вариант № 6

Найти все нечетные числа последовательности Фибоначчи, не превышающие заданного числа А. Последовательность определяется законом:

F0=F1=1; Fn=Fn-1+Fn-2 - для n≥2.

Вариант № 7

Вычислить сумму ряда

∞

S= ∑ (-1)k-1/k c точностью ξ=10-2,…, 10-6. Точное значение: ln 2.

K=1

Определить, как изменяется число итераций при изменении точности.

Вариант № 8

Вычислить: .

Вариант № 9

Вычислить определенный интеграл методом трапеций:

π/3

1. (tg2x+ctg2x) dx при n=20, 30, 40 (число шагов разбиения интервала)

π/6

Вариант № 10

Вычислить сумму ряда

∞

S= 1+∑ (-1)k/(2k)! c точностью ξ=10-2,…, 10-6. Точное значение: cos 1.

K=1

Определить, как изменяется число итераций при изменении точности.

Вариант № 11

Решить задачу, организовав цикл с точностью ξ=10-2,…, 10-6.

Вычислить значение определенного интеграла методом трапеции:

2

1. ex(1+sin x)/(1+cos x) dx при ξ=10-2,…, 10-6.

0

Вариант № 12

Сравнить скорость сходимости (число учитываемых элементов для достижения заданной точности ξ=10-2,…, 10-6) при вычислении числа *e* с помощью ряда и бесконечной дроби:

;

.

Вариант № 13

Сравнить скорость сходимости (число слагаемых для достижения заданной точности ξ=10-2,…, 10-6) следующих разложений числа π:

;

.

Вариант № 14

Для заданных *a* и *p* вычислить  по рекуррентному соотношению Ньютона:



Сколько итераций надо выполнить, чтобы для заданной погрешности ξ=10-2,…, 10-6 выполнялось соотношение: ?

Вариант № 15

Вычислить  для заданного значения a, используя рекуррентное соотношение:



Сколько итераций надо выполнить, чтобы для заданной погрешности ξ=10-2,…, 10-6 выполнялось соотношение: ?

Вариант № 16

Для заданного x>1 вычислить  по итерационной формуле  с заданной погрешностью ξ=10-2,…, 10-6, задав начальное приближение . Сравнить с результатом встроенной функции. Сколько итераций пришлось выполнить?

Вариант № 17

Вычислить, организовав цикл с точностью ξ=10-2,…, 10-6. Определить, как меняется число итераций в зависимости от точности.

.

Вариант № 18

Вычислить, организовав цикл с точностью ξ=10-2,…, 10-6. Определить, как меняется число итераций в зависимости от точности.

.

Вариант № 19

Вычислить, организовав цикл с точностью ξ=10-2,…, 10-6. Определить, как меняется число итераций в зависимости от точности.

.

Вариант № 20

Вычислить, организовав цикл с точностью ξ=10-2,…, 10-8. Определить, как меняется число итераций в зависимости от точности.

.

Вариант № 21

Вычислить, организовав цикл с точностью ξ=10-2,…, 10-8. Определить, как меняется число итераций в зависимости от точности.

.

Вариант № 22

Вычислить, организовав цикл с точностью ξ=10-2,…, 10-8. Определить, как меняется число итераций в зависимости от точности.

,

Вариант № 23

Вычислить, организовав цикл с точностью ξ=10-2,…, 10-8. Определить, как меняется число итераций в зависимости от точности.



Вариант № 24

Вычислить .

Вариант № 25

Вычислить ; *n* слагаемых, *n* вводится с клавиатуры.

Вариант № 26

Вычислить с заданной точностью (ξ=10-2,…, 10-8):

.

Вариант № 27

Вычислить с заданной точностью (ξ=10-2,…, 10-8):

.



## Что нужно знать в результате выполнения работы

1. Понимать механизм работы цикла в программировании.
2. Понимать разницу между циклами с постусловием, предусловием и циклом For.
3. Понимать термины «условие выполнение цикла», «тело цикла» и «итерация цикла».
4. Уметь определять область видимости для переменных.
5. Понимать значение идентификатора const, примененного к переменным.

# Лабораторная работа № 3

Лабораторная работа состоит из двух частей: одномерные массивы и многомерные массивы. Студент выполняет обе части.

В первой части исходные массивы необходимо формировать с помощью генератора случайных чисел, число элементов массива вводится с клавиатуры. Сформированный массив (массивы) необходимо вывести на экран, необходимо также вывести результат работы программы. Используя массив небольшой размерности провести контрольный расчет вручную. *При изменении массивов (сортировка, запись в обратном порядке и др.) новый массив не создавать.* В первой части работы ***необходимо сделать две реализации*** программного обеспечения – ***в первой*** массив должен быть расположен в статической памяти (его размер определяется разработчиком до компиляции кода), а ***во второй*** массив должен быть расположен в динамической памяти (размер массива вводится с клавиатуры при запуске программы). Библиотеку ***STL*** в первой части ***использовать ЗАПРЕЩЕНО***.

Во второй части работы используются двухмерные массивы (матрицы). Матрицы также формируются с помощью генератора случайных чисел, размерности матриц вводятся с клавиатуры. Исходные данные и результаты работы вывести на печать. Для исходных данных небольшой размерность провести ручной расчет для контроля результатов. *При изменении матриц (сортировка строки, запись строки в обратном порядке, транспонирование матрицы и др.) новые массивы (матрицы) не создавать.* Во второй части работы ***массив*** должен быть ***расположен в динамической памяти*** размер массива вводится с клавиатуры при запуске программы). Во второй части работы ***необходимо сделать две реализации*** ПО - ***одна с использованием библиотеки STL*** (std::Vector, std::Array …), и вторая без использования указанной библиотеки.

## Варианты заданий для лабораторной работы № 3. Часть 1

Вариант № 1

Задан одномерный массив целых чисел. Написать программу, которая находит сумму элементов массива.

Вариант № 2

Задан одномерный массив целых чисел. Написать программу, которая находит среднее арифметическое элементов массива.

Вариант № 3

Написать программу, которая находит скалярное произведение двух *n*-мерных векторов.

*Указание*. Скалярное произведение определяется по формуле .

Вариант № 4

Задан одномерный массив целых чисел. Написать программу, которая "переворачивает" массив, т. е. меняет местами его первый и последний элементы, второй и предпоследний и т. д.

Вариант № 5

Заданы целые числа *a1, a2,…, an*. Написать программу, которая находит 

Вариант № 6

Заданы целые числа *a1, a2,…, an*. Написать программу, которая находит .

Вариант № 7

Заданы целые числа *a1, a2,…, an*. Написать программу, которая находит .

Вариант № 8

Заданы целые числа *a1, a2,…, an*. Написать программу, которая находит 

Вариант № 9

Заданы целые числа *a1, a2,…, an*. Написать программу, которая находит количество четных чисел среди чисел *a1, a2,…, an*.

Вариант № 10

Задан одномерный массив целых чисел. Написать программу, которая выводит элементы массива на экран и находит произведение элементов массива.

Вариант № 11

Задан одномерный массив целых чисел. Написать программу, которая выводит элементы массива на экран и находит среднее геометрическое элементов массива.

Вариант № 12

Заданы целые числа *a1, a2,…, an*. Написать программу, которая находит сумму четных чисел среди чисел *a1, a2,…, an*.

Вариант № 13

Заданы целые числа *a1, a2,…, an*. Написать программу, которая находит сумму элементов с четными индексами.

Вариант № 14

Заданы целые числа *a1, a2,…, an*. Написать программу, которая производит сортировку массива по возрастанию.

Вариант № 15

Заданы целые числа *a1, a2,…, an*. Написать программу, которая меняет местами минимальный и максимальный элементы (или любую пару из этих элементов в случае, если таких элементов несколько).

Вариант № 16

Заданы целые числа *a1, a2,…, an*. Написать программу, которая находит индекс минимального элемента.

Вариант № 17

Заданы целые числа *a1, a2,…, an*. Написать программу, которая находит минимальный элемент среди элементов с четными индексами.

Вариант № 18

Заданы целые числа *a1, a2,…, an*. Написать программу, которая находит максимальный четный элемент.

Вариант № 19

Заданы целые числа *a1, a2,…, an*. Написать программу, которая находит сумму отрицательных элементов (при формировании массива обеспечить, чтобы были отрицательные элементы).

Вариант № 20

Заданы целые числа *a1, a2,…, an*. Написать программу, которая считает число нечетных элементов.

Вариант № 21

Заданы целые числа *a1, a2,…, an*. Написать программу, которая считает число элементов кратных 3.

Вариант № 22

Заданы целые числа *a1, a2,…, an*. Написать программу, которая считает сумму элементов кратных 7.

Вариант № 23

Заданы целые числа *a1, a2,…, an*. Написать программу, которая выводит индексы отрицательных элементов на экран (при формировании массива обеспечить, чтобы были отрицательные элементы).

Вариант № 24

Заданы целые числа *a1, a2,…, an*. Число *b* вводится с клавиатуры, проверить, есть ли в массиве элемент со значением *b.* Если элемент есть, то вывести индекс (индексы) такого элемента.

Вариант № 25

Заданы целые числа *a1, a2,…, an*. Написать программу, которая находит количество положительных элементов (при формировании массива обеспечить, чтобы были отрицательные элементы).

Вариант № 26

Заданы целые числа *a1, a2,…, an*. Написать программу, которая отрицательные элементы заменяет на их абсолютные значения и печатает преобразованный массив (при формировании массива обеспечить, чтобы были отрицательные элементы).

Вариант № 27

Заданы целые числа *a1, a2,…, an*. Написать программу, которая нечетные элементы удваивает и печатает преобразованный массив.

## Варианты заданий для лабораторной работы № 3. Часть 2

Вариант № 1

Заданы матрица с элементами размерности *n*×*m* и вектор с элементами размерности *m*. Написать программу, которая находит произведение матрицы на вектор.

*Указание*. Результатом перемножения матрицы на вектор является вектор размерности *n* с компонентами .

Вариант № 2

Заданы две матрицы: размерности *n*×*m* и размерности *m*×*l*. Написать программу, которая находит произведение этих матриц.

*Указание*. Результатом перемножения двух матриц является матрица размерности *n*×*l* с элементами 

Вариант № 3

Задана квадратная матрица размерности *n*×*n*. Написать программу, которая находит транспонированную матрицу.

*Указание*. Транспонированием матрицы называется преобразование, при котором элементы и меняются местами. При транспонировании строки становятся столбцами, а столбцы − строками.

Вариант № 4

Задана действительная матрица *ai*j размерности *n*×*m*. Написать программу, которая определяет числа *b1, b2, …, bn*, равные суммам элементов строк матрицы.

Вариант № 5

Задана целочисленная матрица *ai*j размерности *n*×*m*. Написать программу, которая определяет числа *b1, b2, …, bn*, равные произведениям элементов строк матрицы.

Вариант № 6

Задана действительная матрица *ai*j размерности *n*×*m*. Написать программу, которая определяет числа *b1, b2, …, bn*, равные наименьшим значениям элементов строк матрицы.

Вариант № 7

Задана целочисленная матрица *ai*j размерности *n*×*m*. Написать программу, которая определяет действительные числа *b1, b2, …, bn*, равные значениям средних арифметических элементов строк матрицы.

Вариант № 8

Задана действительная матрица *ai*j размерности *n*×*m*. Написать программу, которая определяет числа *b1, b2, …, bn*, показывающие, сколько отрицательных элементов расположено в каждой строке матрицы.

Вариант № 9

Задана квадратная целочисленная матрица *ai*j размерности *n*×*n*. Написать программу, которая заменяет все отрицательные элементы матрицы на −1, положительные − на 1, а нулевые элементы оставляет без изменения.

Вариант № 10

Задана действительная матрица *ai*j размерности *n*×*m*. Написать программу, которая вычисляет новую матрицу *bij*. Матрица получается путем деления всех элементов заданной матрицы *ai*j на ее наибольший по модулю элемент.

Вариант № 11

Задана целочисленная матрица *ai*j размерности *n*×*m*. Написать программу, которая определяет числа *b1, b2, …, bn*, равные числу нечетных элементов в строках матрицы.

Вариант № 12

Задана действительная матрица *ai*j размерности *n*×*m*. Написать программу, которая определяет действительные числа *b1, b2, …, bn*, равные значениям средних геометрических элементов строк матрицы.

Вариант № 13

Задана действительная матрица *ai*j размерности *n*×*m*. Написать программу, которая определяет действительные числа *b1, b2, …, bn*, равные значениям минимальных элементов строк матрицы.

Вариант № 14

Задана действительная матрица *ai*j размерности *n*×*m*. Написать программу, которая определяет целые числа *b1, b2, …, bn*, равные индексам максимальных элементов строк матрицы (если в строке 2 и более максимальных элементов, то использовать любой индекс).

Вариант № 15

Задана действительная матрица *ai*j размерности *n*×*m*. Написать программу, которая определяет целые числа *b1, b2, …, bn*, равные числу отрицательных элементов строк матрицы (при формировании матрицы обеспечить наличие в ней отрицательных элементов).

Вариант № 16

Улитка. Матрицу M(m, n) заполнить натуральными числами от 1 до *m x n* по спирали, начинающейся в левом верхнем углу и закрученной по часовой стрелке.

Вариант № 17

Работа комбайнера. Матрицу K(m, n) заполнить следующим образом. Элементам, находящимся на периферии (по периметру матрицы), присвоить значение 1; периметру оставшейся подматрицы - значение 2 и так далее до заполнения всей матрицы.

Вариант № 18

Символьную матрицу *A*(*m, n*) заполнить символами с кодами менее 128. Найти строки, содержащие только цифры от 0 до 9, или вывести сообщение, что таких строк нет.

Вариант № 19

Матрицу *A*(*m, n*) заполнить следующим образом. Для заданных *k* и *l* (*k* и *l* вводятся с клавиатуры) элементу *akl* присвоить значение 1; элементам, окаймляющим его (соседним с ним по вертикали, горизонтали и диагоналям) - значение 2; элементам следующего окаймления - значение 3 и так далее до заполнения всей матрицы.

Вариант № 20

Матрица *K*(*m, m*) состоит из нулей и единиц. Найти в ней номера (индексы) хотя бы одной строки или хотя бы одного столбца, не содержащих единицы, либо сообщить, что таковых нет.

Вариант № 21

Латинский квадрат. Латинским квадратом порядка *n* называется квадратная таблица размером, каждая строка и каждый столбец которой содержат все числа от 1 до *n*. Проверить, является ли заданная целочисленная матрица латинским квадратом.

Вариант № 22

Магический квадрат. Магическим квадратом порядка *n* называется квадратная таблица размером *n*, состоящая из чисел так, что суммы по каждому столбцу, каждой строке и каждой из двух диагоналей равны между собой. Проверить, является ли заданная целочисленная квадратная матрица магическим квадратом.

Вариант № 23

Развитие задачи магический квадрат. Реализовать любой алгоритм построения магического квадрата заданного размера *n*.

Вариант № 24

Крестики-нолики. Клеточное поле размером есть результат игры в крестики-нолики на "бесконечном" поле (задать размерность поле равное *n*). Проверить, не закончена ли игра выигрышем "крестиков"? Считается, что "крестики" выиграли, если на поле найдется по горизонтали, вертикали или диагонали цепочка, состоящая подряд из 5 крестиков.

Вариант № 25

В каждой строке матрицы *A*(*n, n*) найти наибольший элемент и поменять его местами с соответствующим диагональным элементом.

Вариант № 26

Матрица *L*(*n, k*) состоит из нулей и единиц. Найти в ней самую длинную цепочку подряд стоящих нулей по горизонтали, вертикали или диагонали.

Вариант № 27

Символьную матрицу *A*(*m, n*) заполнить символами с кодами менее 128. Найти строки, содержащие только английские буквы, или вывести сообщение, что таких строк нет.

## Что нужно знать в результате выполнения работы

1. Понятие указателя.
2. Понимать арифметику указателей.
3. Понятие массива и принцип работы оператора [].
4. Как связаны указатели и массивы.
5. Как массив располагается в памяти.
6. Размещение массивов в статической и динамической памяти.
7. Резервирование памяти под массивы в статической и динамической памяти (создание массивов).
8. Очистка статической и динамической памяти при прекращении использования массива (удаление массивов).
9. Понятие многомерных массивов и их размещение в памяти.
10. Библиотека STL (классы std::vector и std::array).

# Лабораторная работа №4

В заданиях необходимо создать массив структур, число элементов массива заранее неизвестно, вводится с клавиатуры. Значения полей структур также вводятся с клавиатуры. Обратить особое внимание на поля структур, которые сами являются структурами. После ввода массива структур вывести все введенные данные на экран и вычислить требуемые параметры.

При реализации ПО необходимо:

1. Для переменных строкового типа использовать класс String.
2. Для массивов структур использовать библиотеку STL.
3. Для массивов иных типов данных использовать стандартные массивы C++.

## Варианты заданий для лабораторной работы № 4

Варианты 1- 12

Определить структуру «студент», поля структуры: ФИО, номер группы, массив элементов структуры «дисциплина» (не менее 4-х элементов, результаты сдачи сессии), при необходимости можно использовать дополнительные поля (например, число элементов в массиве дисциплин). Структура «дисциплина» включает поля: название, оценка. В программе должно быть минимум две группы и в каждой группе хотя бы по 3 человека.

***Реализуйте дополнительный вариант решения задачи*** с использованием std::map.

В программе задайте перечень студентов (массив переменных структурного типа) и выполните следующие действия (цифра соответствует варианту, то есть каждый вариант выполняет только одно действие).

1. Определите сколько студентов имеют неудовлетворительную оценку хотя бы по одному предмету в каждой группе.
2. Определите сколько студентов сдали все экзамены на 5 в каждой группе.
3. Определите средний балл для всех групп по отдельности по каждой дисциплине.
4. Определите количество отличных оценок, полученных каждой группой по каждой дисциплине.
5. Определите сколько студентов в каждой группе имеют неудовлетворительную оценку по заданному предмету.
6. Определите сколько студентов из каждой группы имеют средний балл от 4 до 5.
7. Определите какое количество неудовлетворительных оценок получено в сумме по всем предметам в каждой группе.
8. Определите какой из предметов был сдан каждой группой лучше всего (по среднему баллу студентов).
9. Определите лучшую группы по среднему баллу студентов (средний балл считается по всем дисциплинам).
10. Подсчитайте средний балл каждой группы по каждой дисциплине и определите, какая группа сдала лучше каждую из дисциплин.
11. Определите студентов с самым высоким средним баллом в каждой группе и на курсе.
12. Составьте рейтинг студентов по среднему баллу.

Варианты 13- 20

Определите структуру «Житель». Поля: ФИО, переменная структурного типа «адрес» (поля: улица, номер дома, номер квартиры), пол, возраст. Определите структуру город (поля: название, массив улиц). В программе должно быть минимум два города, в каждом из которых минимум по три жителя. Также в каждом городе должно совпадать название хотя бы одной улицы (например, в городе 1 есть улица Ленина и в городе 2 тоже есть улица Ленина).

***Реализуйте дополнительный вариант решения задачи*** с использованием std::map.

В программе задайте перечень жителей (массив переменных структурного типа) и выполните следующие действия (цифра соответствует варианту, то есть каждый вариант выполняет только одно действие).

В программе задайте несколько жителей.

1. Определите, сколько лиц женского и сколько мужского пола проживают на каждой улице каждого города.
2. Определите, сколько лиц мужского пола в возрасте старше 18 лет и младше 60 проживают в каждом городе на заданной улице (если улицы в городе нет, то вывести сообщение «В городе X нет улицы Y»).
3. Определите, сколько лиц женского пола в возрасте старше 30 лет проживают в заданном доме.
4. Определите, сколько детей до 7 лет проживают в каждом городе, также укажите на каких улицах они проживают.
5. Определите, сколько лиц мужского пола и женского в возрасте до 50 лет проживают на заданной улице (улица характеризуется названием и городом, в котором находится).
6. Определите, сколько детей от 1 года до 5 проживают в заданном доме (дом характеризуется улицей и городом).
7. Определите, сколько лиц мужского пола призывного возраста (от 18 до 27 лет) находится в базе данных.
8. Определите, число пенсионеров (мужчины с 65 лет, женщины с 60) находится в базе данных.

Варианты 21- 27

Определите структуру «Пассажир» и поместите в нее следующую информацию: ФИО пассажира, массив элементов структуры «багаж» (поля: название предмета багажа, вес предмета в кг), при необходимости можно использовать дополнительные поля (например, число элементов в массиве багажа).

***Реализуйте дополнительный вариант решения задачи*** с использованием std::map.

В программе задайте несколько пассажиров (не менее 6) и выполните следующие действия.

1. Определите число пассажиров, вес багажа которых превышает 30 кг.
2. Определите, имеется ли пассажир, багаж которого состоит из одной вещи весом более чем в 20 кг.
3. Определите средний вес одного предмета багажа для каждого пассажира.
4. Определите количество пассажиров, вес багажа которых превосходит средний.
5. Определите количество пассажиров, имеющих более трех вещей в багаже.
6. Определите средний вес багажа всех пассажиров.
7. Определите пассажира с максимальным весом багажа.

## Что нужно знать в результате выполнения работы

1. Понятие структуры.

2. Создание структуры в статической и динамической памяти.

3. Создание массива структур.

4. Операции для обращения к полям структуры в случае хранения структуры в статической памяти.

5. Операции для обращения к полям структуры в случае хранения структуры в динамической памяти.

6. Понимать, что такое объединения и когда их логично использовать.

7. Понимать, что такое перечисления и когда их логично использовать.

# Лабораторная работа №5

В ходе лабораторной работы необходимо реализовать программу, отвечающую следующим требованиям:

1. Входные данные для программы (за исключением элементов входных массивов) должны передаваться через аргументы командной строки при запуске программы.
2. Все функции, кроме функции main должны быть вынесены в отдельный .cpp файл.
3. В программе должны быть хотя бы одна функция, написанная студентом и осуществляющая вычисления.
4. Реализовать два варианта решения задачи – с использованием рекурсивного вызова функции и без него.

## Варианты заданий для лабораторной работы № 5

Вариант № 1

Решить задачу, используя функцию.

Вычислить N-ый элемент последовательности Фибоначчи.

Вариант № 2

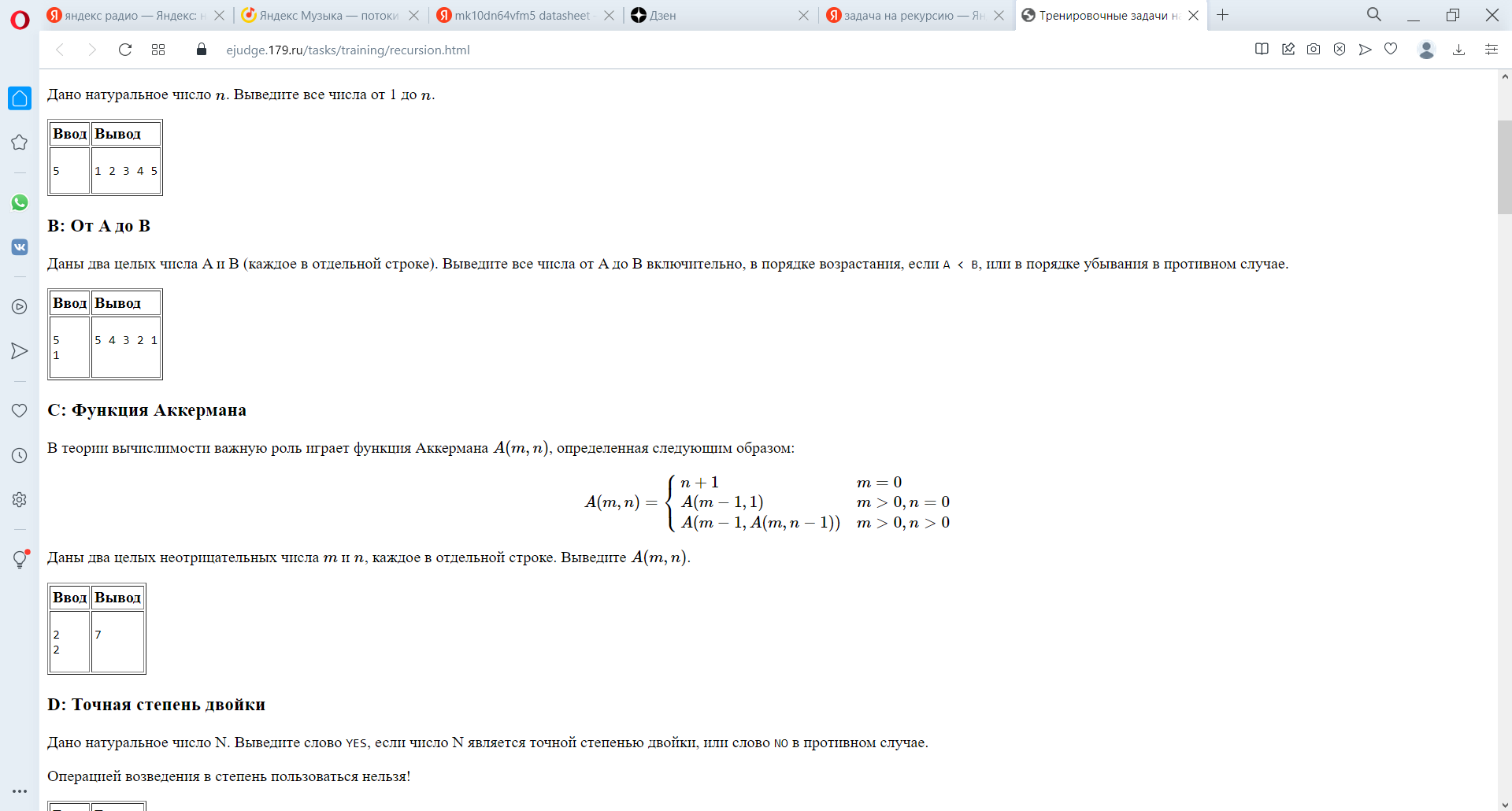
Решить задачу, используя функцию.

Выведите все числа от A до B включительно, в порядке возрастания, если A < B, или в порядке убывания в противном случае.

Вариант № 3

Решить задачу, используя функцию.

Вычислите функцию Аккермана A(m,n), определенную следующим образом:



Вариант № 4

Решить задачу, используя функцию.

Дано натуральное число N. Выведите слово YES, если число N является точной степенью двойки, или слово NO в противном случае. Операцией возведения в степень пользоваться нельзя!

Вариант № 5

Решить задачу, используя функцию.

Дано натуральное число N. Разложите его на слагаемые вида . Операцией деления пользоваться нельзя (а битовыми сдвигами пользоваться рекомендуется).

Вариант № 6

Решить задачу, используя функцию.

Дан массив натуральных чисел длинной N. Отсортируйте массив по возрастанию с использованием сортировки пузырьком.

Вариант № 7

Решить задачу, используя функцию.

Дана символьная матрица размера n\*m. Преобразовать ее таким образом, чтобы элементы каждой нечетной строки расположились в обратном порядке. Новую матрицу не заводить.

Вариант № 8

Решить задачу, используя функцию.

Дано натуральное число N. Вычислите сумму его цифр.

Вариант № 9

Решить задачу, используя функцию.

Дан массив из n натуральных чисел. Определить количество чисел, в десятичной записи которых используется цифра 7.

Вариант№ 10

Решить задачу, используя функцию.

Найдите наибольший общий делитель (НОД) 2-х натуральных чисел по алгоритму Евклида.

*Алгоритм Евклида:*

**1.** Вычислим *r* - остаток от деления числа *a* на *b (a>b), a = bq+r, 0 <= r < b*.

**2.** Если *r = 0*, то *b* есть искомое число (НОД).

**3.** Если *r != 0*, то заменим пару чисел *(a,b)* парой *(b,r),* и перейдем к шагу 1.

Вариант№ 11

Решить задачу, используя функцию.

Дана матрица А(n,m). Определить суммы элементов тех строк матрицы, максимальные элементы которых не превышают среднего значения элементов матрицы или вывести сообщение, что таких строк не существует.

Вариант № 12

Решить задачу, используя функцию.

Дано натуральное число N. Выведите все его цифры по одной, в обратном порядке, разделяя их пробелами.

Вариант № 13

Решить задачу, используя функцию.

Дан массив из n натуральных чисел. Определить количество чисел, десятичная запись которых не содержит одинаковых цифр.

Вариант № 14

Решить задачу, используя функцию.

В целочисленном массиве размерности n найти все числа, у которых старшая значащая десятичная цифра есть 9 (числа сильно различаются по величине).

Вариант № 15

Решить задачу, используя функцию.

Дано натуральное число N. Выведите все его цифры по одной, в обычном порядке, разделяя их пробелами.

Вариант № 16

Решить задачу, используя функцию.

Дан массив натуральных чисел размерности n. Определить число чисел палиндромов в нем. Палиндром («перевертыш») – число, десятичная запись которого читается одинаково слева направо и справа налево (например, 14541).

Вариант № 17

Решить задачу, используя функцию.

Найдите все простые числа < N с использованием решета Эратосфена.

Вариант № 18

Решить задачу, используя функцию.

Дан массив натуральных чисел размерности n. Отсортировать массив в порядке неубывания сумм цифр десятичной записи чисел.

Вариант № 19

Решить задачу, используя функцию.

Дана последовательность натуральных чисел (одно число в строке), завершающаяся числом N. Определите наибольшее значение в этой последовательности. Последовательность вводится по одному символу с клавиатуры.

Вариант № 20

Решить задачу, используя функцию.

Дан массив натуральных чисел размерности n. Отсортировать его в порядке неубывания числа единиц в двоичной записи чисел.

Вариант № 21

Решить задачу, используя функцию.

Дана последовательность натуральных чисел (одно число в строке), завершающаяся числом N. Определите среднее арифметическое значение последовательности. Последовательность вводится по одному символу с клавиатуры.

Вариант № 22

Решить задачу, используя функцию.

Дан массив натуральных чисел размерности n. Вывести на печать все числа этого массива, являющиеся степенями пятерки.

Вариант № 23

Решить задачу, используя функцию.

Дана последовательность натуральных чисел, завершающаяся числом N. Определите значение второго по величине элемента в этой последовательности, то есть элемента, который будет наибольшим, если из последовательности удалить наибольший элемент. Последовательность вводится по одному символу с клавиатуры.

Вариант № 24

Решить задачу, используя функцию.

Дан массив натуральных чисел размерности n. Найти число в массиве, имеющее максимальную сумму цифр в десятичном представлении, если существует несколько чисел с одинаковой максимальной суммой, то вывести на печать их все.

Вариант № 25

Решить задачу, используя функцию.

Дана монотонная последовательность, в которой каждое натуральное число k встречается ровно k раз: 1, 2, 2, 3, 3, 3, 4, 4, 4, 4, ...

По данному натуральному n выведите первые n членов этой последовательности.

Вариант № 26

Решить задачу, используя функцию.

Даны числа a и b. Определите, сколько существует последовательностей из a нулей и b единиц, в которых никакие два нуля не стоят рядом.

Вариант № 27

Решить задачу, используя функцию.

Даны натуральные числа k и s. Определите, сколько существует k-значных натуральных чисел, сумма цифр которых равна s. Запись натурального числа не может начинаться с цифры 0.

## Что нужно знать в результате выполнения работы

1. Объявление и определение функции.
2. Понятие ссылки.
3. L-value и R-value ссылки.
4. Отличие ссылки от указателя.
5. Механизм передачи в функцию параметров по ссылке.
6. Механизм передачи в функцию параметров по указателю.
7. Механизм передачи в функцию массивов и структур.
8. Механизм возвращения из функции массивов и структур.
9. Механизм работы указателей с модификатором const.
10. Понятие рекурсии.
11. Механизм передачи параметров в программу в виде аргументов командной строки.
12. Механизм разработки ПО, состоящего из нескольких .cpp файлов.
13. Понимать, когда к аргументам функции стоит применять идентификатор const.

# Лабораторная работа №6

В лабораторной работе необходимо разработать приложение, работающее в различных режимах: чтения, записи, очистки файла. Также задается текстовый или двоичный режим открытия файла. В режиме записи исходные данные вводятся с клавиатуры (количество объектов, заданных структурным типом, данные этих объектов), введенные данные ***дозаписываются*** в файл, не затирая предыдущую информацию. В режиме чтения данные, сохраненные в режиме записи в файле, читаются из файла и выводятся на консоль. В режиме очистки файла производится удаление всей информации из файла (по окончании работы данного режима файл на ПК должен существовать и не иметь содержимого).

Режимы приложения задаются как параметры командной строки и сохраняются в глобальные переменные: вначале чтение или запись, затем текстовый или двоичный режим, далее имя файла, например:

write binary MyFile.dat режим записи, файл в двоичном режиме, имя файла MyFile.dat;

read text MyFile.txt режим чтения, файл в текстовом режиме, имя файла MyFile.txt.

Также в ПО необходимо обрабатывать ситуацию, при которой читаемый файл оказался битым, т.е. в нем хранятся некорректные данные, например, поле типа int хранит строковое значение.

ПО должно содержать несколько осмысленных функций, разработанных студентом.

***Необходимо использовать библиотеку fstream.***

## Варианты заданий для лабораторной работы №6

В работе создать массив (вектор) объектов структурного типа размерности n, необходимо сохранить его в файле в текстовом или двоичном режимах, обеспечить чтение данных из файла. Сделать выводы о сравнительных размерах двух файлов с данными (текстовом и двоичном). Использовать в работе шаблон vector и класс string. Режимы приложения задаются как параметры командной строки.

Варианты заданий заданы в таблице.

Таблица 1 – Варианты заданий

|  |  |
| --- | --- |
| **Варианты** | **Параметры приложений** |
| **1, 13** | Структура «студент» (поля: ФИО, группа, номер зачетной книжки, массив 4-х оценок за сессию) |
| **2, 14** | Структура «подразделение организации» (поля: название, ФИО начальника, число сотрудников, число сотрудников в отпуске) |
| **3, 15** | Структура «автомобиль» (поля: модель, массив расходов топлива из 3-х элементов (на трассе, в городе, смешанный), максимальная скорость, мощность) |
| **4, 16** | Структура «сотрудник» (поля: ФИО, дата приема на работу, должность, базовый оклад) |
| **5, 17** | Структура «банковский вклад» (поля: название, сумма вклада, тип валюты, ставка в % годовых) |
| **6, 18** | Структура «компьютер» (поля: тип процессора, тактовая частота, объем оперативной памяти, размер жесткого диска, тип видеокарты) |
| **7, 19** | Структура «программное средство криптозащиты» (поля: название разработчика, номер версии, название реализуемого алгоритма, наличие лицензии ФСБ- да, нет) |
| **8, 20** | Структура «товар в магазине (на складе)» (поля: название, число единиц в наличии, стоимость одной единицы) |
| **9, 21** | Структура «специализированная школа» (поля: город, номер, специализация, число учеников) |
| **10, 22** | Структура «автобус» (поля: модель (название), длина, высота, максимальное число пассажиров) |
| **11, 23** | Структура «пассажирский самолет» (поля: модель (название), максимальная высота полета, максимальная скорость полета, максимальная дальность, максимальное число пассажиров) |
| **12, 24** | Структура «океанское судно» (поля: название, ФИО капитана, водоизмещение в тыс. тонн, численность экипажа) |

## Что нужно знать в результате выполнения лабораторной работы

1. Режимы открытия файлов (должен ли файл существовать перед открытием на чтение, всегда ли?).
2. Способ записи в файл одной переменной и массива данных.
3. Способ чтения из файла одной переменной и массива данных.
4. Способ вычисления размера данных.
5. Способ чтения данных, начиная с определенной позиции в файле.

# Лабораторная работа №7

В лабораторной работе необходимо организовать список объектов и сортировку списка. Данные списка читаются из файла, путь к файлу указывается в качестве аргумента командной строки. При сортировке элементы списка остаются в оперативной памяти на «своих местах», меняются только значения указателей, связывающие элементы. Вывести на экран список до сортировки и после сортировки.

В работе ***ЗАПРЕЩЕНО*** использовать списки из библиотеки STL. Список необходимо реализовывать самостоятельно с использованием указателей.

## Варианты заданий для лабораторной работы №7

В приложении организовать список объектов и сортировку списка. Для задания строк и массивов использовать шаблоны string и vector.

Варианты заданий заданы в ячейках таблицы 1. Параметры приложений указаны в заголовках строк и столбцов таблицы 1.

Таблица 1 – Варианты заданий

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Параметры приложений** | | **Метод сортировки** | | |
| **Метод вставки** | **Метод прямого выбора** | **Метод обмена («пузырька»)** |
| Объект- сотрудник (поля: ФИО, дата приема на работу, должность, базовый оклад) | Сортировка по ФИО, двухсвязный кольцевой список | 1 | 8 | 18 |
| Сортировка по окладу, двухсвязный некольцевой список | 4 | 5 | 6 |
| Объект- банковский вклад (поля: название, сумма вклада, тип валюты, ставка в % годовых) | Сортировка по сумме вклада, двухсвязный кольцевой список | 7 | 2 | 9 |
| Сортировка по названию, двухсвязный некольцевой список | 10 | 11 | 12 |
|  |  |  |  |  |
| Объект- студент (поля: ФИО, группа, номер зачетной книжки, массив 4-х оценок за сессию) | Сортировка по ФИО, двухсвязный кольцевой список | 13 | 14 | 15 |
| Сортировка по среднему баллу, двухсвязный некольцевой список | 16 | 17 | 3 |
| Объект- книга (поля: ФИО автора, название, издательство, год издания, число страниц) | Сортировка по ФИО, двухсвязный кольцевой список | 19 | 20 | 21 |
| Сортировка по названию, двухсвязный некольцевой список | 22 | 23 | 24 |
| Объект- автомобиль (поля: название модели, массив расходов топлива из 3-х элементов (на трассе, в городе, смешанный), максимальная скорость, мощность) | Сортировка по названию модели, двухсвязный кольцевой список | 25 | 26 | 27 |
| Сортировка по смешанному расходу топлива, двухсвязный некольцевой список | 28 | 29 | 30 |

## Что нужно знать в результате выполнения лабораторной работы

1. Понимать механизм работы указателей.
2. Понимать размещение списка в памяти.
3. Понимать, что такое линейный односвязный список.
4. Понимать, что такое линейный кольцевой односвязный список.
5. Понимать, что такое линейный двухсвязный список.
6. Понимать, что такое линейный двухсвязный линейный кольцевой список.
7. Понимать алгоритмы сортировки методом вставки, методом прямого выбора и методом пузырька.
8. Понимать, чем отличаются глобальные и локальные переменные.
9. Понимать, что означает ключевое слово static.

# Лабораторная работа №8

В ходе лабораторной работы студент производит улучшение своих навыков владения контейнерами библиотеки STL. Лабораторная работа состоит из двух частей (необходимо разработать два экземпляра ПО).

В первой части лабораторной работы необходимо провести вычислительные эксперименты по оценки времени выполнения некоторых операций с контейнерами. Необходимо измерять время выполнения заданных операций для пары контейнеров в соответствии со своим вариантом. Для измерения времени можно использовать класс LogDuration, определенный в файле profile.h. Файл можно скачать по ссылке:

[http://student.bmstu.cloud:10288/AlgWeb\_2022\_1/mydata/ZadForSem/profile.h](http://student.bmstu.cloud:10288/AlgWeb_2021_1/mydata/ZadForSem/profile.h)

Во второй части лабораторной работы необходимо определить структуру в соответствии с заданным вариантов, создать заданный контейнер переменных этого структурного типа. Выполнить сортировку объектов в контейнере, использую алгоритм sort или для контейнера list метод класса sort. Для сортировки использовать специальную функцию предикат, см. пример ниже сортировки вектора по разным полям. Распечатать контейнер объектов до сортировки и после сортировки.

Ввод исходных данных осуществлять из текстового файла (не менее 8 объектов), тестовый файл создать отдельно в простом текстовом редакторе типа «Блокнот», отдельно обрабатывать ситуацию некорректного заполнения файла. Путь к файлу передавать в виде аргумента командной строки.

Пример сортировки контейнера с использование функции предиката:

#include <iostream>

#include <vector>

#include <string>

#include <algorithm>

using namespace std;

struct Test

{

string name;

double value;

};

// Сортировка по строковому полю (функция - предикат)

bool ForSortString(const Test & ob1, const Test & ob2)

{

return ob1.name < ob2.name;

}

// Сортировка по вещественному полю (функция - предикат)

bool ForSortDouble(const Test & ob1, const Test & ob2)

{

return ob1.value < ob2.value;

}

// Печать вектора

void print(const vector<Test>& vec)

{

for (auto & pos : vec)

cout << "name=" << pos.name << " value=" << pos.value << endl;

}

int main()

{

vector<Test> vec;

vec.push\_back({ "Hello", 45.23 });

vec.push\_back({ "Yes", 3.233 });

vec.push\_back({ "World", 12 });

vec.push\_back({ "Not", 10 });

vec.push\_back({ "I", 13 });

vec.push\_back({ "An", 18 });

print(vec);

sort(vec.begin(), vec.end(), ForSortString); // Передается указатель на функцию

cout << "\nSorted by string:\n";

print(vec);

sort(vec.begin(), vec.end(), ForSortDouble); // Передается указатель на функцию

cout << "\nSorted by double:\n";

print(vec);

return 1;

}

## Варианты заданий для лабораторной работы №8

В работе число выполняемых вызовов функций (итераций циклов, число итераций может быть порядка: 100000, 1000000, 10000000) студент должен подобрать сам с учетом быстродействия компьютера, таким образом, чтобы время выполнения кода было разумным (интервалы измерялись миллисекундами, десятками или сотнями миллисекунд, секундами, в крайнем случае, десятками секунд). Необходимо произвести одинаковое число аналогичных операций (вызовов функций) с двумя сравниваемыми контейнерами. Измерения проводить отдельно по каждому из типов операций. Заполнять контейнер, а также задавать элементы для поиска и удаления, когда это требуется, необходимо с помощью генератора псевдослучайных чисел, типы элементов контейнера – целые числа.

Варианты заданий приведены в таблице.

|  |  |  |
| --- | --- | --- |
| № варианта | Сравниваемые контейнеры | Исследуемые операции |
|  | vector и list | 1 Добавление элементов в конец контейнера.  2. Поиск элементов.  3. Удаление элементов из начала контейнера. |
|  | vector и list | 1 Добавление элементов в начало контейнера.  2. Поиск элементов.  3. Удаление элементов из конца контейнера. |
|  | vector и deque | 1 Добавление элементов в конец контейнера.  2. Поиск элементов.  3. Удаление элемента из начала контейнера. |
|  | vector и deque | 1 Добавление элементов в начало контейнера.  2. Поиск элементов.  3. Удаление элементов из конца контейнера. |
|  | list и deque | 1 Добавление элементов в конец контейнера.  2. Поиск элементов.  3. Удаление элементов из начала контейнера. |
|  | list и deque | 1 Добавление элементов в начало контейнера.  2. Поиск элементов.  3. Удаление элементов из конца контейнера. |
|  | vector и set | 1 Добавление элементов в конец vector и вставка в set.  2. Поиск элементов.  3. Удаление элементов из начала vector и удаление из set. |
|  | list и set | 1 Добавление элементов в конец list и вставка в set.  2. Поиск элементов.  3. Удаление элемента из начала list и удаление из set. |
|  | deque и set | 1 Добавление элементов в конец deque и вставка в set.  2. Поиск элементов.  3. Удаление элементов из начала deque и удаление из set. |
|  | vector и unordered\_set | 1 Добавление элементов в конец vector и вставка в unordered\_set.  2. Поиск элементов.  3. Удаление элементов из начала vector и удаление из unordered\_set. |
|  | list и unordered\_set | 1 Добавление элементов в конец list и вставка в unordered\_set.  2. Поиск элементов.  3. Удаление элементов из начала list и удаление из unordered\_set. |
|  | deque и unordered\_set | 1 Добавление элементов в конец deque и вставка в unordered\_set.  2. Поиск элементов.  3. Удаление элементов из начала deque и удаление из unordered\_set. |
|  | set и unordered\_set | 1 Вставка элементовt.  2. Поиск элементов.  3. Удаление элементов. |
|  | vector и set | 1 Добавление элементов в начало vector и вставка в set.  2. Поиск элементов.  3. Удаление элементов из конца vector и удаление из set. |
|  | list и set | 1 Добавление элементов в начало list и вставка в set.  2. Поиск элементов.  3. Удаление элемента из конца list и удаление из set. |
|  | deque и set | 1 Добавление элементов в начало deque и вставка в set.  2. Поиск элементов.  3. Удаление элементов из конца deque и удаление из set. |
|  | vector и unordered\_set | 1 Добавление элементов в начало vector и вставка в unordered\_set.  2. Поиск элементов.  3. Удаление элементов из конца vector и удаление из unordered\_set. |
|  | list и unordered\_set | 1 Добавление элементов в начало list и вставка в unordered\_set.  2. Поиск элементов.  3. Удаление элементов из конца list и удаление из unordered\_set. |
|  | deque и unordered\_set | 1 Добавление элементов в начало deque и вставка в unordered\_set.  2. Поиск элементов.  3. Удаление элементов из конца deque и удаление из unordered\_set. |
|  | vector и set | 1 Добавление элементов в начало vector и вставка в set.  2. Поиск элементов.  3. Удаление элементов из начала vector и удаление из set. |
|  | list и set | 1 Добавление элементов в начало list и вставка в set.  2. Поиск элементов.  3. Удаление элементов из начала list и удаление из set. |
|  | deque и set | 1 Добавление элементов в начало deque и вставка в set.  2. Поиск элементов.  3. Удаление элементов из начала deque и удаление из set. |
|  | vector и unordered\_set | 1 Добавление элементов в начало vector и вставка в unordered\_set.  2. Поиск элементов.  3. Удаление элементов из начала vector и удаление из unordered\_set. |
|  | list и unordered\_set | 1 Добавление элементов в начало list и вставка в unordered\_set.  2. Поиск элементов.  3. Удаление элементов из начала list и удаление из unordered\_set. |
|  | deque и unordered\_set | 1 Добавление элементов в начало deque и вставка в unordered\_set.  2. Поиск элементов.  3. Удаление элементов из начала deque и удаление из unordered\_set. |

## Что нужно знать в результате выполнения лабораторной работы

1. Типы контейнеров в библиотеке STL (vector, list, deque, set, unordered set, map, array).
2. Понимать отличия между разными типами контейнеров.
3. Знать основные операции по работе с контейнерами.
4. Функции предикаты и указатели на функции.