

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Оренбургский государственный университет»

Кафедра промышленной электроники
и информационно-измерительной техники

С. А. Сильвашко

ПРАКТИКУМ ПО ПРОГРАММИРОВАНИЮ НА C++

Методические указания

Рекомендовано к изданию редакционно-издательским советом федерального государственного бюджетного образовательного учреждения высшего образования «Оренбургский государственный университет» для обучающихся по образовательным программам высшего образования по направлениям подготовки 11.03.02 Инфокоммуникационные технологии и системы связи, 11.03.03 Конструирование и технология электронных средств и 11.03.04 Электроника и наноэлектроника

Оренбург
2019

УДК 004.42(076.5)
ББК 32.973.26-018я7
С 36

Рецензент – доцент, кандидат технических наук А. В. Хлуденев

Сильвашко, С. А.
С 36 Практикум по программированию на С++: методические указания / С. А. Сильвашко; Оренбургский гос. ун-т. – Оренбург : ОГУ, 2019. – 32 с.

В методических указаниях изложены задания для выполнения лабораторных работ и практических занятий, предусмотренных рабочей программой дисциплины «Информационные технологии в электронике, радиотехнике и системах связи».

Методические указания предназначены для обучающихся по программам высшего образования по направлениям подготовки 11.03.02 Инфокоммуникационные технологии и системы связи, 11.03.03 Конструирование и технология электронных средств и 11.03.04 Электроника и наноэлектроника

УДК 004.42(076.5)
ББК 32.973.26-018я7

© Сильвашко С. А., 2019
© ОГУ, 2019

Содержание

Введение	4
1 Практические занятия	5
1.1 Практическое занятие № 1. Освоение среды программирования Microsoft Visual C++	5
1.2 Практическое занятие № 2. Программирование линейных алгоритмов	6
1.3 Практическое занятие № 3. Программирование циклических алгоритмов	9
1.4 Практическое занятие № 4. Создание оконного приложения в среде Microsoft Visual C++	11
2 Лабораторные работы	13
2.1 Лабораторная работа № 1. Программирование разветвляющихся алгоритмов ..	13
2.2 Лабораторная работа № 2. Работа с массивами	16
2.3 Лабораторная работа № 3. Освоение элементов объектно-ориентированного программирования	19
2.4 Лабораторная работа № 4. Использование комплексных чисел в программе на C++	23
3 Литература, рекомендуемая для изучения при подготовке к занятиям	26
Приложение А Варианты исходных данных	27

Введение

Язык программирования С++ является одним из наиболее популярных языков, используемых в объектно-ориентированном программировании. Он является развитием языка С (Си), широко используемого при программировании микроконтроллеров, и поддерживает операторы этого языка. С учетом этого специалисту в области современной электроники целесообразно владеть навыками программирования на языке С++.

Практикум содержит задания на практические занятия и лабораторные работы, предусмотренные при изучении дисциплины «Информационные технологии в электронике, радиотехнике и системах связи», и направленные на формирование у обучающихся навыков алгоритмизации и программирования простейших инженерных задач.

Методические указания предназначены для обучающихся по образовательным программам высшего образования по направлениям подготовки 11.03.02 Инфокоммуникационные технологии и системы связи, 11.03.03 Конструирование и технология электронных средств и 11.03.04 Электроника и наноэлектроника

1 Практические занятия

1.1 Практическое занятие № 1. Освоение среды программирования Microsoft Visual C++

Целью занятия является освоение интерфейса интегрированной среды разработки программ MS Visual C++ 2010 Express, освоение инструментальных средств среды разработки.

Аудиторное время, отводимое на занятие – 2 академических часа.

1.1.1 Задание для подготовки к практическому занятию

1.1.1.1 Изучить материал учебного пособия [2, С. 6 – 22] по теме практического занятия.

1.1.2 Задание на практическое занятие

1.1.2.1 Запустить программу Microsoft Visual C++ 2010 Express и ознакомиться с содержимым окна программы. Ознакомиться с командами главного меню окна программы.

1.1.2.2 С помощью контекстного меню свойств окна обозревателя решений исследовать, как изменяет вид окна назначение каждого из возможных свойств.

1.1.2.3 Выполнить действия, предусмотренные п. 1.1.2.2, в отношении окна «*Начальная страница*». Сравнить результаты выполнения п. 1.1.2.2 и п. 1.1.2.3.

1.1.2.4 Создать консольное приложение CLR на основе шаблона, в котором предусмотреть вывод в окне программы *cmd.exe* текстового фрагмента «**Я <Фамилия Имя>, студент второго курса, приступил к изучению основ программирования на языке C++**». Проект сохранить в папке на рабочем столе пользователя (название папки и файла – **фамилия исполнителя**). Создать выполняемый файл.

1.1.2.5 Подготовить отчет и представить руководителю занятия. В отчет перенести изображение введенного в редакторе кода текста приложения и вид окна программы *cmd.exe* с результатом выполнения разработанной программы.

1.1.3 Примерные контрольные вопросы

1.1.3.1 Какие компоненты включает интегрированная среда разработки программ Visual C++ 2010 Express?

1.1.3.2 Как открыть диалоговое окно со свойствами окон программы Visual C++ 2010 Express?

1.1.3.3 Как открыть существующий проект на C++ в Visual C++ 2010 Express?
Как создать новый проект?

1.1.3.4 Какие виды проектов можно создать в среде Visual C++ 2010 Express?

1.1.3.5 Как создать консольное приложение в среде Visual C++ 2010 Express?

1.1.3.6 Как запустить на выполнение консольное приложение?

1.1.3.7 Где отображается информация об ошибках, выявленных в ходе компиляции программы?

1.1.3.8 Как найти обнаруженную компилятором ошибку в тексте программы?

1.1.3.9 Как обеспечить задержку на экране монитора окна консоли после запуска приложения?

1.1.3.10 Как обеспечить вывод текста в консоли на русском языке?

1.2 Практическое занятие № 2. Программирование линейных алгоритмов

Целью занятия является приобретение навыков составления алгоритмов линейных вычислительных процессов и их реализации на языке высокого уровня C++.

Аудиторное время, отводимое на занятие – 4 академических часа.

1.2.1 Задание для подготовки к практическому занятию

1.2.1.1 Изучить материал учебного пособия [2, С. 23 – 51] по теме практического занятия.

1.2.1.2 Разработать схему алгоритма вычислительного процесса решения задачи с учетом варианта индивидуальных заданий (таблица 1).

1.2.2 Задание на практическое занятие

1.2.2.1 Разработать консольное приложение на языке C++ для проведения вычислений в соответствии с заданием (таблица 1).

Для ввода исходных данных и вывода результатов вычислений использовать объекты-потоки `cin` и `cout`. **Текст программы сопроводить комментариями.**

Таблица 1 – Варианты индивидуальных заданий

Номера вариантов	Задание
1	Вычислить площадь круга и длину окружности диаметром D .
2	Вычислить площадь боковой поверхности и объем кругового прямого цилиндра.
3	Вычислить площадь полной поверхности усеченного кругового цилиндра.
4	Вычислить площадь полной поверхности и объем кругового прямого конуса.
5	Вычислить площадь боковой поверхности и объем усеченного прямого конуса.
6	Вычислить площадь основания и объем прямоугольного параллелепипеда.
7	Вычислить площадь полной поверхности и периметр основания прямоугольного параллелепипеда.
8	Вычислить площадь боковой поверхности и объем правильной пирамиды с треугольным основанием
9	Вычислить площадь боковой поверхности и объем правильной пирамиды с прямоугольным основанием
10	Вычислить площадь поверхности и объем сферы.
11	Вычислить площадь полной поверхности усеченного кругового цилиндра.
12	Вычислить объем цилиндрической трубы диаметром D с толщиной стенки d .
13	Вычислить площадь боковой поверхности и объем правильной призмы с треугольным основанием
14	Вычислить площадь полной поверхности и объем правильной призмы с прямоугольным основанием
15	Вычислить объем прямой треугольной призмы, усеченной не параллельно основанию
16	Вычислить объем усеченной правильной пирамиды с треугольным основанием (плоскость сечения параллельна основанию)
17	Вычислить площадь основания и полной поверхности усеченной правильной пирамиды с треугольным основанием (плоскость сечения параллельна основанию)
18	Вычислить площадь боковой поверхности и объем усеченной правильной пирамиды с прямоугольным основанием (плоскость сечения параллельна основанию)
19	Вычислить площадь боковой поверхности и объем шарового сегмента, высота которого равна радиусу
20	Вычислить площадь полной поверхности и объем тора

Примечания

1 В качестве исходных данных, вводимых с клавиатуры, служат параметры геометрической фигуры (высота, ширина, образующая и т. п.).

2 Полагать, что параметры геометрической фигуры могут иметь, в том числе, и вещественные значения.

1.2.2.2 Подготовить отчет и представить руководителю занятия.

Отчет должен содержать схему алгоритма, текст программы и вид окна консольного приложения с результатами выполнения программы.

1.2.3 Примерные контрольные вопросы

1.2.3.1 Какие символы можно использовать в программе на C++?

1.2.3.2 Что называют управляющими последовательностями? Приведите примеры.

1.2.3.3 Что называется идентификатором? Как правильно записать идентификатор переменной в C++?

1.2.3.4 Приведите примеры знаков операций? Какие различают виды операций с учетом количества участвующих в них операндов?

1.2.3.5 Поясните особенности использования операции деления (/).

1.2.3.6 Что называется комментарием? Как записать комментарий в программе?

1.2.3.7 Какие типы переменных различают в C++?

1.2.3.8 Приведите формат и примеры оператора присваивания.

1.2.3.9 Как выполнить ввод исходных данных с клавиатуры при выполнении программы?

1.2.3.10 Как вывести информацию на экран монитора (на консоль)?

1.2.3.11 Назовите основные операторы, используемые при программировании линейного алгоритма.

1.2.3.12 Поясните сущность объявления и инициализации переменной в программе. Приведите формат инструкции объявления переменной.

1.2.3.13 Можно ли в одной инструкции объявить сразу несколько переменных?

1.2.3.14 Опишите структуру программы на C++.

1.2.3.15 Что такое директива препроцессора?

1.3 Практическое занятие № 3. Программирование циклических алгоритмов

Целью занятия является приобретение навыков составления алгоритмов циклических вычислительных процессов и их реализации на языке высокого уровня C++.

Аудиторное время, отводимое на занятие – 4 академических часа.

1.3.1 Задание для подготовки к практическому занятию

1.3.1.1 Изучить материал учебного пособия [2, С. 57 – 62, 75 – 82] по теме практического занятия.

1.3.1.2 Разработать схемы алгоритмов вычислительных процессов решения задач с учетом варианта индивидуальных заданий.

1.3.2 Задание на практическое занятие

1.3.2.1 Разработать консольное приложение для вычисления:

– суммы $s = \sum_{i=0}^{5+n} (n+i)$, где n – четные порядковые номера обучающихся в

списке учебной группы;

– произведения $p = \prod_{i=1}^{5+n} (n+i)$, где n – нечетные порядковые номера обучаю-

щихся в списке учебной группы.

1.3.2.2 Разработать консольное приложение для вычисления значений функции, заданной графически (таблицы А.1 и А.2), на временном интервале, превышающем

длительность импульса.

Решение вывести в окне консоли в виде таблицы, содержащей заглавия столбцов и разделительные линии между отдельными столбцами данных.

Предусмотреть вывод результатов вычислений в текстовый файл.

1.3.2.3 Подготовить отчет и представить руководителю занятия.

Отчет должен содержать разработанные алгоритмы, тексты программ, вид окон консольных приложений при выполнении программ, окно текстового редактора с содержимым сохраненного файла.

1.3.3 Примерные контрольные вопросы

1.3.3.1 Какой вычислительный процесс называется циклическим?

1.3.3.2 Приведите схему алгоритма цикла с параметром.

1.3.3.3 Приведите схему алгоритма цикла с предусловием.

1.3.3.4 Приведите схему алгоритма цикла с постусловием.

1.3.3.5 Приведите и поясните формат оператора цикла **for**.

1.3.3.6 Приведите и поясните формат оператора цикла **while**.

1.3.3.7 Приведите и поясните формат оператора цикла **do while**.

1.3.3.8 Приведите условия предпочтительного использования того или иного оператора цикла.

1.3.3.9 Какие циклы называют вложенными? Поясните правила программирования вложенных циклов.

1.3.3.10 Как сохранить результаты вычислений в файл?

1.3.3.11 Как открыть файл для записи в него данных?

1.3.3.12 Как добавить данные в существующий файл?

1.3.3.13 Как открыть файл для ввода из него данных в программу?

1.4 Практическое занятие № 4. Создание оконного приложения в среде Microsoft Visual C++

Целью занятия является овладение навыками разработки оконных приложений с помощью конструктора форм Windows Forms.

Аудиторное время, отводимое на занятие – 8 академических часов.

1.4.1 Задание для подготовки к практическому занятию

1.4.1.1 Изучить материал рекомендованной литературы [1] по теме практического занятия.

1.4.1.2 Продумать структуру окна программы (расположение областей ввода данных, кнопок и т. п.).

1.4.2 Задание на практическое занятие

1.4.2.1 Разработать оконное приложение на C++, с помощью которого по введенным значениям параметров импульсного сигнала U_m и τ (временная диаграмма сигнала представлена в таблице А.2, параметры сигнала – в таблице А.1), а также аргумента t , вычисляется мгновенное значение напряжения сигнала. Аргумент функции t (текущее время) может принимать значения, в том числе, за пределами интервала существования импульса.

Предусмотреть вывод сообщения об ошибке при неправильном вводе исходных данных (если введена буква вместо цифры или точка вместо запятой в вещественном числе).

1.4.2.2 Подготовить отчет и представить руководителю занятия.

Отчет должен содержать текст программы и вид окна приложения для двух случаев: правильно и неправильно введенные исходные данные.

1.4.3 Примерные контрольные вопросы

1.4.3.1 Перечислите основные этапы создания оконного приложения с помощью шаблона *Приложение Windows Forms*.

1.4.3.2 На какой панели размещены компоненты графического интерфейса пользователя (элементы управления)?

1.4.3.3 Каким способом разместить требуемые элементы управления в экранную форму проекта?

1.4.3.4 Как изменить свойства элемента управления?

1.4.3.5 Что понимают под событием, связанным с элементом управления?

1.4.3.6 Что понимают под процедурой обработки события?

1.4.3.7 Что такое обработчик события?

1.4.3.8 Как добавить в программный код «пустой» обработчик события?

1.4.3.9 Для чего предназначен элемент управления *PictureBox*?

1.4.3.10 Как можно изменить размер графической поверхности элемента управления *PictureBox*?

1.4.3.11 Чем отличается *стандартный набор карандашей* от *системного набора карандашей*?

1.4.3.12 Для чего предназначен элемент управления *Textbox*?

1.4.3.13 Как следует изменить свойства элемента управления *Textbox*, чтобы в текстовом поле можно было выводить больше одной строки?

1.4.3.14 Как добавить в элементе управления *Textbox* вертикальную (горизонтальную) полосу прокрутки?

2 Лабораторные работы

2.1 Лабораторная работа № 1. Программирование разветвляющихся алгоритмов

Целью лабораторной работы является приобретение навыков составления алгоритмов разветвляющихся вычислительных процессов и их реализации на языке высокого уровня C++.

Аудиторное время, отводимое на занятие – 4 академических часа.

2.1.1 Задание для подготовки к лабораторной работе

2.1.1.1 Изучить материал учебного пособия [2, С. 51 – 57] по теме лабораторной работы.

2.1.1.2 Разработать схемы алгоритмов вычислительных процессов решения задач с учетом варианта индивидуальных заданий.

2.1.2 Задание на лабораторную работу

2.1.2.1 Приобрести навыки алгоритмизации и программирования разветвляющихся вычислительных процессов на языке высокого уровня C++.

2.1.3 Порядок выполнения лабораторной работы

2.1.3.1 Разработать консольное приложение для определения, попадает ли точка M с координатами (x, y) в круг радиусом r , центр которого расположен:

- 1) в начале координат;
- 2) в точке, смещенной вдоль оси x на расстояние d от начала координат.

2.1.3.2 Разработать консольное приложение для решения задачи с учетом индивидуального задания (таблица 2).

2.1.3.3 Разработать консольное приложение, с помощью которого вычисляется значение функции, заданной графически (варианты заданий сведены в таблицах А.1 и А.2), в текущий момент времени t .

В качестве исходных данных вводить параметры сигнала (U_m и τ) и текущее время t .

Таблица 2 – Варианты индивидуальных заданий

Номера вариантов	Задания
1	Написать программу вычисления сопротивления участка электрической цепи, состоящего из двух сопротивлений. Сопротивления могут быть соединены последовательно или параллельно.
2	Написать программу вычисления эквивалентной емкости при соединении двух конденсаторов. Конденсаторы могут быть соединены последовательно или параллельно.
3	Написать программу для вычисления частного от деления двух чисел. Предусмотреть вывод сообщения об ошибке в случае равенства нулю знаменателя.
4	Написать программу вычисления стоимости покупки с учетом скидки. Скидка в 10% предоставляется, если сумма покупки больше 500 руб.
5	Написать программу, которая сравнивает два введенных с клавиатуры числа. Программа должна указать, какое число больше. Если числа равны, то вывести соответствующее сообщение.
6	Написать программу, которая проверяет, является ли введенное пользователем целое число четным и выводит соответствующее сообщение.
7	Написать программу, которая проверяет, делится ли введенное с клавиатуры целое число на три нацело и выводит соответствующее сообщение.
8	Написать программу, которая вычисляет оптимальный вес для пользователя, сравнивает его с реальным и выдает рекомендацию, на сколько килограммов ему следует поправиться или похудеть. Оптимальный вес вычисляется по формуле: $\text{рост (см)} - 100$.
9	Написать программу, которая запрашивает у пользователя номер месяца и выводит соответствующее название времени года. В случае, если пользователь введет недопустимое число, программа выводит сообщение: «Ошибка ввода данных».
10	Написать программу, которая запрашивает у пользователя номер дня недели и выводит одно из сообщений: «Рабочий день», «Суббота» или «Воскресенье».
11	Написать программу, которая запрашивает у пользователя значения сопротивлений резисторов и вид схемы усилителя на ОУ (инвертирующий или неинвертирующий), а затем выдает значение коэффициента усиления.
12	Написать программу, которая запрашивает у пользователя три действительных числа, а затем возводит в квадрат неотрицательные из этих чисел и в четвертую степень – отрицательные.
13	Даны две точки $A(x_1, y_1)$ и $B(x_2, y_2)$. Написать программу, определяющую, какая из этих точек находится ближе к началу координат.
14	Даны два не равных действительных числа. Разработать программу, заменяющую меньшее из этих двух чисел половиной их суммы, а большее – их удвоенным произведением.
15	Даны два целых числа. Разработать программу, реализующую алгоритм: если числа не равны, заменить меньшее из них большим исходных числом, а если равны, то заменить оба исходных числа нулями.
16	Разработать программу, определяющую результат гадания на ромашке (любит – не любит), используя в качестве исходных данных – число лепестков n .
17	Создать программную модель анализа пожарного датчика в помещении и вывода сообщения «Пожароопасная ситуация», если температура (вводится с клавиатуры) превышает 60°C .

Продолжение таблицы 2

Номера вариантов	Задания
18	Разработать программу, позволяющую по введенному с клавиатуры возрасту человека отнести его к одной из четырех групп: дошкольник, ученик, работник, пенсионер.
19	Разработать программу, определяющую максимальное число среди трех введенных с клавиатуры чисел (с учетом знака числа).
20	Написать программу вычисления стоимости покупки с учетом скидки. Скидка в 3% предоставляется, если сумма покупки больше 300 руб., в 5% – если сумма больше 500 руб.

2.1.3.4 Подготовить отчет и представить руководителю занятия.

Отчет должен содержать разработанные алгоритмы, тексты программ и вид окон консольных приложений при выполнении программ.

2.1.4 Примерные контрольные вопросы

2.1.4.1 Какой вычислительный процесс называется разветвляющимся?

2.1.4.2 Приведите схему разветвляющегося алгоритма.

2.1.4.3 Поясните различия разветвляющихся алгоритмов с полным и неполным выбором.

2.1.4.4 Приведите схему алгоритма ВЫБОР.

2.1.4.5 Приведите формат условного оператора **if**.

2.1.4.6 Приведите список операций отношения, используемых в условном операторе **if**.

2.1.4.7 Приведите список логических операций, используемых в условном операторе **if**.

2.1.4.8 В каком случае в разветвляющемся вычислительном процессе используют оператор **switch**?

2.1.4.9 Приведите формат оператора **switch**.

2.1.4.10 С помощью каких операторов можно организовать выход из переключателя **switch**?

2.2 Лабораторная работа № 2. Работа с массивами

Целью лабораторной работы является приобретение навыков программирования на языке высокого уровня C++ задач, связанных с обработкой массивов данных.

Аудиторное время, отводимое на занятие – 4 академических часа.

2.2.1 Задание для подготовки к лабораторной работе

2.2.1.1 Изучить материал учебного пособия [2, С. 67 – 82] по теме лабораторной работы.

2.2.2 Задание на лабораторную работу

2.2.2.1 Приобрести навыки разработки программ на языке высокого уровня C++ для обработки данных, представленных в виде массива.

2.2.3 Порядок выполнения лабораторной работы

2.2.3.1 Разработать консольное приложение для решения задачи с учетом варианта индивидуальных заданий (таблица 3).

Таблица 3 – Варианты индивидуальных заданий

Номера вариантов	Задания
1	В одномерном массиве, состоящем из n вещественных элементов: 1) вычислить сумму отрицательных элементов массива; 2) вычислить произведение элементов массива, расположенных между максимальным и минимальным элементами; 3) упорядочить элементы массива по возрастанию.
2	В одномерном массиве, состоящем из n вещественных элементов: 1) вычислить сумму положительных элементов массива; 2) вычислить произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами; 3) упорядочить элементы массива по убыванию.
3	В одномерном массиве, состоящем из n целых элементов: 1) вычислить произведение элементов массива с четными номерами; 2) вычислить сумму элементов массива, расположенных между первым и последним нулевыми элементами;

Продолжение таблицы 3

Номера вариантов	Задания
	3) преобразовать массив таким образом, чтобы сначала располагались все положительные элементы, а потом – все отрицательные (элементы, равные 0, считать положительными).
4	В одномерном массиве, состоящем из n вещественных элементов: 1) вычислить сумму элементов массива с нечетными номерами; 2) вычислить сумму элементов массива, расположенных между первым и последним отрицательными элементами; 3) сжать массив, удалив из него все элементы, модуль которых не превышает 1. Освободившиеся в конце массива элементы заполнить нулями.
5	В одномерном массиве, состоящем из n вещественных элементов: 1) вычислить максимальный элемент массива; 2) вычислить сумму элементов массива, расположенных до последнего положительного элемента; 3) сжать массив, удалив из него все элементы, модуль которых находится в интервале $[a, b]$. Освободившиеся в конце массива элементы заполнить нулями.
6	В одномерном массиве, состоящем из n вещественных элементов: 1) вычислить минимальный элемент массива; 2) вычислить сумму элементов массива, расположенных между первым и последним положительными элементами; 3) преобразовать массив таким образом, чтобы сначала располагались все элементы, равные нулю, а потом – все остальные.
7	В одномерном массиве, состоящем из n целых элементов: 1) вычислить номер максимального элемента массива; 2) вычислить произведение элементов массива, расположенных между первым и вторым нулевыми элементами; 3) преобразовать массив таким образом, чтобы в первой его половине располагались элементы, стоявшие в нечетных позициях, а во второй половине – элементы, стоявшие в четных позициях.
8	В одномерном массиве, состоящем из n вещественных элементов: 1) вычислить номер минимального элемента массива; 2) вычислить сумму элементов массива, расположенных между первым и вторым отрицательными элементами; 3) преобразовать массив таким образом, чтобы сначала располагались все элементы, модуль которых не превышает 1, а потом – все остальные.
9	В одномерном массиве, состоящем из n вещественных элементов: 1) вычислить максимальный по модулю элемент массива; 2) вычислить сумму элементов массива, расположенных между первым и вторым положительными элементами; 3) преобразовать массив таким образом, чтобы элементы, равные нулю, располагались после всех остальных.
10	В одномерном массиве, состоящем из n целых элементов: 1) вычислить минимальный по модулю элемент массива; 2) вычислить сумму модулей элементов массива, расположенных после первого элемента, равного нулю; 3) преобразовать массив таким образом, чтобы в первой его половине располагались элементы, стоявшие в четных позициях, а во второй половине – элементы, стоявшие в нечетных позициях.

Продолжение таблицы 3

Номера вариантов	Задания
11	<p>В одномерном массиве, состоящем из n вещественных элементов:</p> <ol style="list-style-type: none"> 1) вычислить номер минимального по модулю элемента массива; 2) вычислить сумму модулей элементов массива, расположенных после первого отрицательного элемента; 3) сжать массив, удалив из него все элементы, величина которых находится в интервале $[a, b]$. Освободившиеся в конце массива элементы заполнить нулями.
12	<p>В одномерном массиве, состоящем из n вещественных элементов:</p> <ol style="list-style-type: none"> 1) вычислить номер максимального по модулю элемента массива; 2) вычислить сумму элементов массива, расположенных после первого положительного элемента; 3) преобразовать массив таким образом, чтобы сначала располагались все элементы, целая часть которых лежит в интервале $[a, b]$, а потом – все остальные.
13	<p>В одномерном массиве, состоящем из n вещественных элементов:</p> <ol style="list-style-type: none"> 1) вычислить количество элементов массива, лежащих в диапазоне от A до B; 2) вычислить сумму элементов массива, расположенных после максимального элемента; 3) упорядочить элементы массива по убыванию модулей элементов.
14	<p>В одномерном массиве, состоящем из n вещественных элементов:</p> <ol style="list-style-type: none"> 1) вычислить количество элементов массива, равных 0; 2) вычислить сумму элементов массива, расположенных после минимального элемента; 3) упорядочить элементы массива по возрастанию модулей элементов.
15	<p>В одномерном массиве, состоящем из n вещественных элементов:</p> <ol style="list-style-type: none"> 1) вычислить количество элементов массива, больших C; 2) вычислить произведение элементов массива, расположенных после максимального по модулю элемента; 3) преобразовать массив таким образом, чтобы сначала располагались все отрицательные элементы, а потом – все положительные (элементы, равные 0, считать положительными).
16	<p>В одномерном массиве, состоящем из n вещественных элементов:</p> <ol style="list-style-type: none"> 1) вычислить количество отрицательных элементов массива; 2) вычислить сумму модулей элементов массива, расположенных после минимального по модулю элемента; 3) заменить все отрицательные элементы массива их квадратами и упорядочить элементы массива по возрастанию.
17	<p>В одномерном массиве, состоящем из n целых элементов:</p> <ol style="list-style-type: none"> 1) вычислить количество положительных элементов массива; 2) вычислить сумму элементов массива, расположенных после последнего элемента, равного нулю; 3) преобразовать массив таким образом, чтобы сначала располагались все элементы, целая часть которых не превышает 1, а затем – все остальные.
18	<p>В одномерном массиве, состоящем из n вещественных элементов:</p> <ol style="list-style-type: none"> 1) вычислить количество элементов массива, меньших C; 2) вычислить сумму целых частей элементов массива, расположенных после последнего отрицательного элемента; 3) преобразовать массив таким образом, чтобы сначала располагались все элементы, отличающиеся от максимального не более чем на 20%, а затем – все остальные.

Продолжение таблицы 3

Номера вариантов	Задания
19	В одномерном массиве, состоящем из n вещественных элементов: 1) вычислить произведение отрицательных элементов массива; 2) вычислить сумму положительных элементов массива, расположенных до максимального элемента; 3) изменить порядок следования элементов в массиве на обратный.
20	В одномерном массиве, состоящем из n вещественных элементов: 1) вычислить произведение положительных элементов массива; 2) вычислить сумму элементов массива, расположенных до минимального элемента; 3) упорядочить по возрастанию отдельно элементы, стоящие на четных местах, и элементы, стоящие на нечетных местах.

2.2.3.2 Подготовить отчет и представить руководителю занятия.

Отчет должен содержать текст программы и вид окна консольного приложения при выполнении программы с проверкой решения всех предусмотренных заданий.

2.2.4 Примерные контрольные вопросы

2.2.4.1 Что понимают под массивом?

2.2.4.2 Приведите формат объявления одномерного и многомерного массивов.

2.2.4.3 Как выполнить инициализацию массива?

2.2.4.4 С какого номера начинается индексация элементов массива?

2.2.4.5 Какой массив называют динамическим?

2.2.4.6 Как объявить динамический массив?

2.2.4.7 Как освободить память, зарезервированную под элементы динамического массива?

2.3 Лабораторная работа № 3. Освоение элементов объектно-ориентированного программирования

Целью лабораторной работы является освоение элементов объектно-ориентированного программирования языка высокого уровня C++.

Аудиторное время, отводимое на занятие – 4 академических часа.

2.3.1 Задание для подготовки к лабораторной работе

2.3.1.1 Изучить материал учебного пособия [2, С. 85 – 90] по теме лабораторной работы.

2.3.1.2 Продумать структуру описания класса и структуру меню программы для выбора методов класса с учетом варианта индивидуального задания.

2.3.2 Задание на лабораторную работу

2.3.2.1 Ознакомиться с основными идеями объектно-ориентированного программирования на языке высокого уровня C++ и приобрести первичные навыки создания собственных классов.

2.3.3 Порядок выполнения лабораторной работы

2.3.3.1 Составить описание класса объектов с учетом варианта индивидуального задания (таблица 4).

Таблица 4 – Варианты индивидуальных заданий

Номера вариантов	Класс	Методы класса
1	Плоские геометрические фигуры: квадрат, прямоугольник	Создание объектов, изменение размеров, вращение на заданный угол
2	Плоские геометрические фигуры: окружность, эллипс	Создание объектов, перемещение на плоскости, изменение размеров
3	Информация о почтовом адресе организации	Раздельное изменение составных частей адреса, создание и уничтожение объектов класса
4	Комплексные числа	Выполнение операций сложения, вычитания и умножения комплексных чисел
5	Комплексные числа	Выполнение операций сложения комплексных чисел и вычисления модуля и аргумента комплексного числа
6	Векторы, задаваемые координатами на плоскости	Выполнение операций сложения и вычитания векторов, вычисления скалярного произведения двух векторов
7	Прямоугольники со сторонами, параллельными осям координат	Перемещение прямоугольников на плоскости, изменение размеров, построение наименьшего прямоугольника, содержащего два заданных прямоугольника

Продолжение таблицы 4

Номера вариантов	Класс	Методы класса
8	Одномерные массивы целых чисел	Возможность обращения к отдельному элементу массива с контролем выхода за пределы массива, возможность выполнения операций поэлементного сложения и вычитания массивов с одинаковыми границами индексов
9	Одномерные массивы вещественных чисел	Возможность умножения и деления всех элементов массива на скаляр, вывода на экран элемента массива по заданному индексу, вывода на экран всего массива
10	Многочлены от одной переменной, задаваемые степенью многочлена и массивом коэффициентов	Вычисление значения многочлена для заданного аргумента, операции сложения и умножения многочленов с получением нового объекта-многочлена, вывод на экран его описания
11	Многочлены от одной переменной, задаваемые степенью многочлена и массивом коэффициентов	Вывод на экран описания многочленов, операции вычитания многочленов с получением нового объекта-многочлена, умножения многочлена на константу
12	Матрицы произвольного размера	Возможность изменения числа строк и столбцов, вывода на экран подматрицы любого размера и всей матрицы
13	Домашняя библиотека	Возможность добавления книг в библиотеку, удаления книг из нее, поиска книги по какому-либо признаку (например, по автору или по году издания)
14	Записная книжка	Возможность работы с произвольным числом записей, добавления и удаления записей, поиска записи по какому-либо признаку (например, по фамилии, дате рождения или номеру телефона)
15	Телефонный справочник	Возможность работы с локальными и федеральными номерами, добавления и удаления записей, поиска номера по какому-либо признаку (например, по фамилии, имени-отчеству или по двум последним цифрам)
16	Студенческая группа	Возможность работы с переменным числом студентов, добавления и удаления записей, поиска студента по какому-либо признаку (например, по фамилии, дате рождения или номеру телефона)
17	Вещественные матрицы	Возможность реализовывать следующие операции над матрицами: сложение, вычитание, умножение (матрицы на матрицу), деление (матрицы на число)
18	Легковые автомобили	Возможность добавления и удаления объектов класса, поиска объектов по марке, цвету кузова, типу двигателя (инжекторный, дизельный), мощности двигателя
19	Вещественные матрицы	Возможность реализовывать следующие операции над матрицами: вычисления обратной и транспонированной матрицы, поиска наименьшего и наибольшего элемента матрицы
20	Радиодетали	Возможность добавления и удаления объектов класса, вывода на экран количества элементов по выбранному типу и номиналу

2.3.3.2 Разработать консольное приложение, демонстрирующее работу с этим классом. Программа должна содержать меню, позволяющее осуществить проверку всех методов класса.

2.3.3.3 Подготовить отчет и представить руководителю занятия.

Отчет должен содержать текст программы и вид окон консольного приложения при выполнении программы с проверкой реализации предусмотренных методов класса.

2.3.4 Примерные контрольные вопросы

2.3.4.1 В чем состоят различия между модульным и объектно-ориентированным программированием?

2.3.4.2 Перечислите этапы разработки программ в объектно-ориентированном стиле.

2.3.4.3 Что понимают под классом в C++?

2.3.4.4 Назовите базовые принципы объектно-ориентированного программирования.

2.3.4.5 Поясните сущность инкапсуляции.

2.3.4.6 В чем сущность наследования?

2.3.4.7 В чем сущность полиморфизма?

2.3.4.8 Приведите структуру описания класса.

2.3.4.9 Поясните различия между глобальными и локальными классами.

2.3.4.10 Что такое экземпляр класса?

2.3.4.11 Каким образом осуществляется доступ к элементам данных и методам класса или объекта?

2.3.4.12 С помощью каких методов выполняется создание и удаление объектов?

2.3.4.13 Назовите статусы доступа к компонентам класса.

2.3.4.14 Как обеспечивается доступ из основной программы к методам класса?

2.4 Лабораторная работа № 4. Использование комплексных чисел в программе на C++

Целью лабораторной работы является освоение элементов объектно-ориентированного программирования языка высокого уровня C++.

Аудиторное время, отводимое на занятие – 4 академических часа.

2.4.1 Задание для подготовки к лабораторной работе

2.4.1.1 Изучить материал учебного пособия [2, С. 90 – 94] по теме лабораторной работы.

2.4.1.2 Составить выражение для комплексного коэффициента передачи напряжения четырехполюсника (рисунок 1) в соответствии с вариантом индивидуального задания.

Примечание – Обобщенные элементы схемы четырехполюсника (рисунок 1) $Z1$ и $Z2$ в каждом варианте задания заменять соответствующими элементами (или последовательным соединением элементов) в соответствии с таблицей 5.

2.4.2 Задание на лабораторную работу

2.4.2.1 Приобрести навыки использования в программах на C++ функций класса комплексных чисел.

2.4.3 Порядок выполнения лабораторной работы

2.4.3.1 Разработать консольное приложение для вычисления значений амплитудно-частотной (АЧХ) и фазочастотной (ФЧХ) характеристик четырехполюсника на основе математического выражения, полученного при подготовке к занятию.

Результаты вычислений вывести в виде двух таблиц (отдельно АЧХ и ФЧХ) и в файлы с именами **koeff.txt** (АЧХ) и **phase.txt** (ФЧХ).

Диапазон изменения циклической частоты (в герцах) принять равным $f \in [0, 1000]$, шаг изменения частоты – 50 Гц.

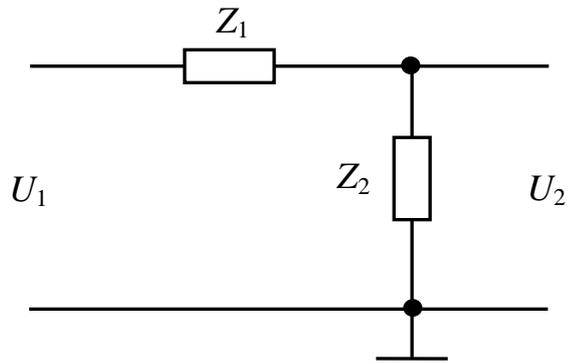


Рисунок 1 – Схема четырехполюсника

Таблица 5 – Варианты индивидуальных заданий

Номера вариантов	Z_1	Z_2
1	C_1	R_2
2	R_1	C_2
3	$C_1 + R_1$	R_2
4	C_1	$C_2 + R_2$
5	R_1	$C_2 + R_2$
6	$C_1 + R_1$	C_2
7	L_1	R_2
8	R_1	L_2
9	$L_1 + R_1$	R_2
10	R_1	$L_2 + R_2$
11	C_1	R_2
12	R_1	C_2
13	$C_1 + R_1$	R_2
14	C_1	$C_2 + R_2$
15	R_1	$C_2 + R_2$
16	$C_1 + R_1$	C_2
17	L_1	R_2
18	R_1	L_2
19	$L_1 + R_1$	R_2
20	R_1	$L_2 + R_2$

Таблица 6 – Номиналы элементов четырехполюсника

Номера вариантов	R_1 , Ом	C_1 , мкФ	L_1 , Гн	R_2 , Ом	C_2 , мкФ	L_2 , Гн
1	-	1,0	-	1500	-	-
2	2200	-	-	-	1,0	-
3	1000	2,2	-	1000	-	-
4	-	1,0	-	1000	3,3	-
5	1000	-	-	2000	1,0	-
6	1000	1,0	-	-	3,3	-

Продолжение таблицы 6

Номера вариантов	$R1$, Ом	$C1$, мкФ	$L1$, Гн	$R2$, Ом	$C2$, мкФ	$L2$, Гн
7	-	-	2,0	1000	-	-
8	1000	-	-	-	-	1,0
9	1200	-	3,3	1200	-	-
10	1200	-	-	510	-	2,0
11	-	2,2	-	510	-	-
12	1000	-	-	-	3,3	-
13	510	1,0	-	1000	-	-
14	-	0,22	-	2000	2,2	-
15	1800	-	-	1800	0,33	-
16	2000	0,47	-	-	2,2	-
17	-	-	4,0	1500	-	-
18	2200	-	-	-	-	3,0
19	1200	-	2,0	510	-	-
20	510	-	-	510	-	2,2

2.2.3.2 Подготовить отчет и представить руководителю занятия.

Отчет должен содержать текст программы, вид окна консольного приложения с результатами выполнения программы, виды окон текстового редактора с открытыми файлами **koeff.txt** и **phase.txt**.

2.4.4 Примерные контрольные вопросы

2.4.4.1 Приведите формат объявления комплексной переменной.

2.4.4.2 Какие операторы можно использовать в выражениях с комплексными числами?

2.4.4.3 С помощью каких функций можно вычислить модуль и аргумент комплексного числа?

2.4.4.4 Как выполнить инициализацию комплексной переменной?

2.4.4.5 Как извлечь действительную часть комплексного числа?

2.4.4.6 Как извлечь мнимую часть комплексного числа?

3 Литература, рекомендуемая для изучения при подготовке к занятиям

1 Сильвашко, С. А. Объектно-ориентированное программирование [Электронный ресурс] : метод. указания к курсовому проектированию / С. А. Сильвашко; М-во образования и науки Рос. Федерации, Федер. гос. бюджет. образоват. учреждение высш. образования «Оренбург. гос. ун-т», Каф. пром. электроники и информ.-измер. техники. – Электрон. текстовые дан. (1 файл: 0.77 Мб). – Оренбург : ОГУ, 2017. – 28 с. – Загл. с тит. экрана. – Adobe Acrobat Reader 6.0.

2 Сильвашко, С. А. Основы программирования микроконтроллеров на C++ [Электронный ресурс] : учебное пособие / С. А. Сильвашко; Оренбургский гос. ун-т. – Оренбург : ОГУ, 2019. – 126 с.

3 Слабнов, В. Д. Программирование на C++ [Электронный ресурс] : лекции / В. Д. Слабнов; Институт экономики, управления и права (г. Казань). – Казань : Познание, 2012. – 136 с. – ISBN 978-5-8399-0386-9. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=364222>. – ЭБС «Университетская библиотека онлайн».

4 Павловская, Т. А. C/C++. Программирование на языке высокого уровня : для магистров и бакалавров: учебник для студентов высших учебных заведений / Т. А. Павловская. – Санкт-Петербург : Питер, 2013. – 461 с. – ISBN 978-5-496-00031-4.

Приложение А

(обязательное)

Варианты исходных данных

Таблица А.1 – Параметры импульсных сигналов

Номера вариантов	Амплитуда импульса U , В	Длительность импульса τ_u , мс
1	10	0,1
2	5	0,1
3	7	0,4
4	4	0,6
5	8	0,5
6	1	0,4
7	6	0,7
8	2	0,8
9	3	0,6
10	9	0,8
11	3	0,5
12	4	0,1
13	6	0,4
14	7	0,6
15	5	0,2
16	5	0,4
17	3	0,8
18	10	0,5
19	6	0,4
20	4	0,1

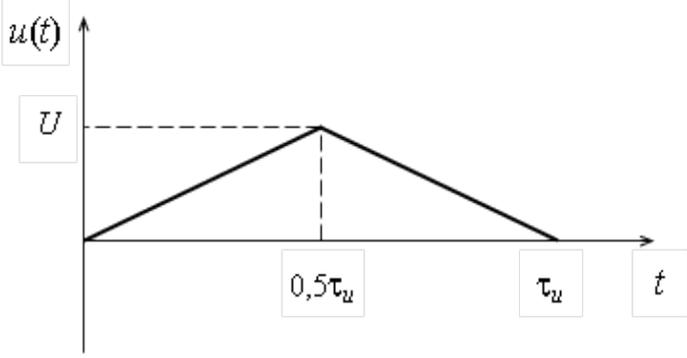
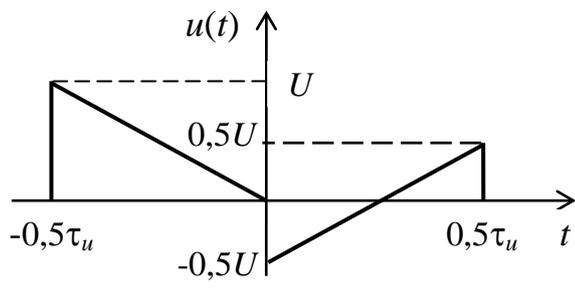
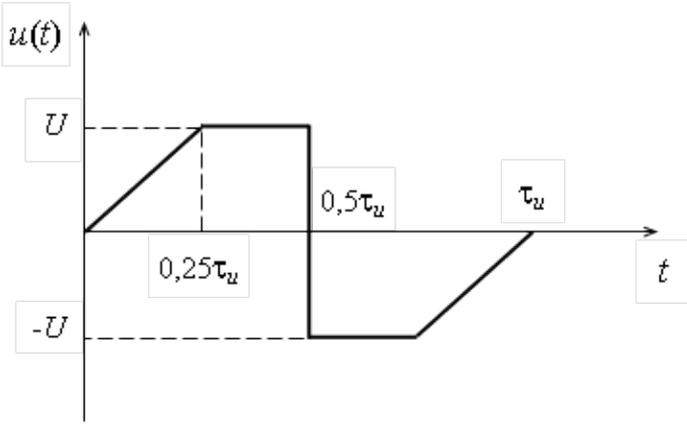
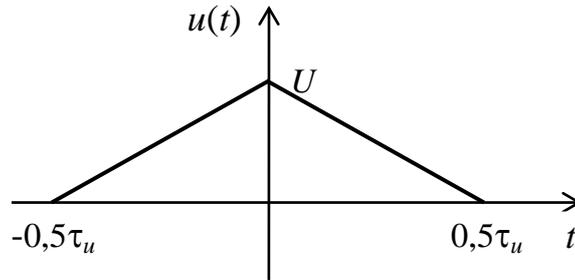
Таблица А.2 – Временные диаграммы сигналов

Номера вариантов	Закон изменения напряжения
1	<p>Graph 1: Voltage $u(t)$ vs time t. The signal starts at 0, rises linearly from $t=0.2\tau_u$ to $t=0.4\tau_u$ to a peak value U, remains constant at U until $t=0.6\tau_u$, falls linearly to 0 at $t=0.8\tau_u$, and remains at 0 until $t=\tau_u$.</p>
2	<p>Graph 2: Voltage $u(t)$ vs time t. The signal starts at $-0.2U$, jumps to U at $t=0.4\tau_u$, remains constant at U until $t=0.8\tau_u$, jumps to $-0.2U$, and remains constant at $-0.2U$ until $t=\tau_u$.</p>
3	<p>Graph 3: Voltage $u(t)$ vs time t. The signal starts at 0, rises linearly to U at $t=0.25\tau_u$, falls linearly to $-U$ at $t=0.75\tau_u$, and rises linearly back to 0 at $t=\tau_u$.</p>
4	<p>Graph 4: Voltage $u(t)$ vs time t. The signal starts at U, drops to $-U$ at $t=0.5\tau_u$, and remains constant at $-U$ until $t=\tau_u$.</p>

Продолжение таблицы А.4

Номера вариантов	Закон изменения напряжения
5	<p>Graph showing voltage $u(t)$ versus time t. The voltage starts at U at $t = -0,5\tau_u$, decreases linearly to $-U$ at $t = 0,5\tau_u$, and then jumps back to U.</p>
6	<p>Graph showing voltage $u(t)$ versus time t. The voltage is U from $t = -0,5\tau_u$ to $t = 0$, then decreases linearly to $-U$ at $t = 0$, and increases linearly back to U at $t = 0,5\tau_u$.</p>
7	<p>Graph showing voltage $u(t)$ versus time t. The voltage is 0 until $t = 0,2\tau_u$, then jumps to U until $t = 0,4\tau_u$, then jumps to $-U$ until $t = 0,8\tau_u$, and finally jumps back to 0.</p>
8	<p>Graph showing voltage $u(t)$ versus time t. The voltage increases linearly from 0 to U at $t = 0,5\tau_u$, then jumps to $-U$, and finally increases linearly back to 0 at $t = \tau_u$.</p>

Продолжение таблицы А.4

Номера вариантов	Закон изменения напряжения
9	 <p>Graph showing voltage $u(t)$ versus time t. The voltage starts at 0, rises linearly to a peak value U at time $0,5\tau_u$, and then falls linearly to 0 at time τ_u.</p>
10	 <p>Graph showing voltage $u(t)$ versus time t. The voltage starts at 0, rises to U at $-0,5\tau_u$, falls to $0,5U$ at $t=0$, then falls to $-0,5U$ at $0,5\tau_u$, and finally rises to 0 at τ_u.</p>
11	 <p>Graph showing voltage $u(t)$ versus time t. The voltage starts at 0, rises to U at $0,25\tau_u$, stays constant at U until $0,5\tau_u$, drops to $-U$, stays constant at $-U$ until $0,75\tau_u$, and then rises to 0 at τ_u.</p>
12	 <p>Graph showing voltage $u(t)$ versus time t. The voltage starts at 0, rises linearly to U at $t=0$, and then falls linearly to 0 at $0,5\tau_u$.</p>

Продолжение таблицы А.4

Номера вариантов	Закон изменения напряжения
13	
14	
15	
16	

Продолжение таблицы А.4

Номера вариантов	Закон изменения напряжения
17	
18	
19	
20	