

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Комсомольский-на-Амуре государственный университет»

Кафедра «Математическое обеспечение и применение ЭВМ»

УТВЕРЖДАЮ
Первый проректор
И.В. Макурин
12 2017 г.



РАБОЧАЯ ПРОГРАММА

дисциплины «Программирование на языке высокого уровня»

основной профессиональной образовательной программы
подготовки бакалавров

по направлению 09.03.01 «Информатика и вычислительная техника»
профиль «Программное обеспечение средств вычислительной техники
и автоматизированных систем»

Форма обучения	Заочная
Технология обучения	Традиционная

Комсомольск-на-Амуре 2017

Автор рабочей программы
старший преподаватель

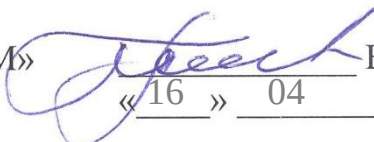
 С. Ю. Александров
« 15 » 04 2017 г.

СОГЛАСОВАНО

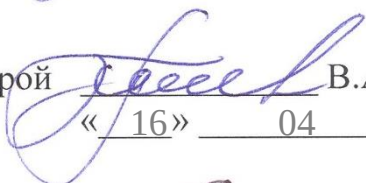
Директор библиотеки

 И. А. Романовская
« 18 » 04 2017 г.

Заведующий кафедрой «МОП ЭВМ»

 В. А. Тихомиров
« 16 » 04 2017 .

Заведующий выпускающей кафедрой
«МОП ЭВМ»

 В. А. Тихомиров
« 16 » 04 2017 .

/ Декан факультета заочного и дистанци-
онного обучения, кандидат технических
наук, доцент

 М. В. Семибратова
« 19 » 04 2017 г.

Начальник учебно-методического
управления

 Е. Е. Поздеева
« 22 » 04 2017 г.

Введение

Рабочая программа дисциплины «Программирование на языке высокого уровня» составлена в соответствии с требованиями федерального государственного образовательного стандарта, утвержденного приказом Министерства образования и науки Российской Федерации от 12.01.2016 № 5, и основной профессиональной образовательной программы подготовки бакалавров по направлению 09.03.01 «Информатика и вычислительная техника».

1 Аннотация дисциплины

Наименование дисциплины	Программирование на языке высокого уровня											
Цель дисциплины	Научить студентов применению языков высокого уровня для решения типовых задач											
Задачи дисциплины	Дать представление студентам о возможностях современных высокоуровневых языков программирования. Познакомить с подходами к решению типовых задач по представлению и обработке информации.											
Основные разделы дисциплины	Процедурное программирование Объектно-ориентированное программирование Библиотеки классов и фреймворки											
Общая трудоемкость дисциплины	12 з.е. / 432 академических часа											
	Семестр	Аудиторная нагрузка, ч				СРС, ч	Промежуточная аттестация, ч	Всего за семестр, ч				
		Лекции	Пр. занятия	Лаб. работы	Курсовое проектирование							
		2	4		8					128	4	144
		3	6		8					157	9	180
	4	4		6		94	4	108				
ИТОГО:		14		22		379	17	432				

2 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами образовательной программы

Дисциплина «Программирование на языке высокого уровня» нацелена на формирование компетенций, знаний, умений и навыков, указанных в таблице 1.

Таблица 1 – Компетенции, знания, умения, навыки

Наименование и шифр компетенции, в формировании которой принимает участие дисциплина	Перечень формируемых знаний, умений, навыков, предусмотренных образовательной программой		
	Перечень знаний (с указанием шифра)	Перечень умений (с указанием шифра)	Перечень навыков (с указанием шифра)
Способность разрабатывать компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии программирования (ПК-2)	Знать возможности существующей программно-технической архитектуры З1(ПК-2-1)	Уметь проводить оценку и обоснование рекомендуемых решений при разработке ПО У1(ПК-2-1)	Владеть приемами разработки ПО на языках разного уровня Н1(ПК-2-1)
	Знать типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения З1(ПК-2-2)	Уметь применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов У1(ПК-2-2)	Владеть навыками использования библиотек программных модулей, шаблонов, классов объектов при разработке ПО Н1(ПК-2-2)
	Знать возможности существующей программно-технической архитектуры З1(ПК-2-3)	Уметь проводить оценку и обоснование рекомендуемых решений при разработке ПО У1(ПК-2-3)	Владеть навыками разработки ПО на базе классов Н1(ПК-2-3)

3 Место дисциплины в структуре образовательной программы

Дисциплина «Программирование на языке высокого уровня» изучается на 1 и 2 курсе во 2, 3 и 4 семестрах.

Дисциплина входит в состав блока 1 «Дисциплины (модули)» и относится к обязательным дисциплинам вариативной части учебного плана.

С изучения дисциплины «Программирование на языке высокого уровня» начинается второй этап освоения компетенции ПК-2: способность разрабатывать компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии программирования.

Знания, умения и навыки, полученные в ходе изучения разделов данной дисциплины, потребуются на всех последующих этапах освоения компетенции ПК-2.

Входной контроль проводится в виде тестирования. Задания тестов представлены в приложении А РПД.

4 Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Общая трудоемкость (объем) дисциплины составляет 12 зачетных единиц, 432 академических часа.

Распределение объема дисциплины (модуля) по видам учебных занятий представлено в таблице 2.

Таблица 2 – Объем дисциплины (модуля) по видам учебных занятий

Объем дисциплины	Всего академических часов
	Заочная форма обучения
Общая трудоемкость дисциплины	432
Контактная аудиторная работа обучающихся с преподавателем (по видам учебных занятий), всего	36
В том числе:	
занятия лекционного типа (лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации педагогическими работниками)	14
занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия)	22
Самостоятельная работа обучающихся и контактная работа , включающая групповые консультации, индивидуальную работу обучающихся с преподавателями (в том числе индивидуальные консультации); взаимодействие в электронной информационно-образовательной среде вуза	379
Промежуточная аттестация обучающихся	17

5 Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

Таблица 3 – Структура и содержание дисциплины (модуля)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)	Форма проведения	Планируемые (контролируемые) результаты освоения	
				Компетенции	Знания, умения, навыки
Раздел 1 <i>Процедурное программирование</i>					
Тема 1. Базовые средства языка Состав языка. Типы данных. Структура программы. Описание переменных. Простейшие средства ввода-вывода. Выражения. Преобразования базовых типов. Основные операторы. Составные типы данных в стиле C	Лекция	2	Традиционная	ПК-2	31(ПК-2-1)
Тема 2. Модульное программирование Функции в C++. Функции библиотеки языка C. Директивы препроцессора. Области действия и пространства имен.	Лекция	1	Традиционная	ПК-2	31(ПК-2-1)
Тема 3. Введение в организацию данных Абстрактные структуры данных. Динамические структуры данных.	Лекция	1	Активная	ПК-2	31(ПК-2-1)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)	Форма проведения	Планируемые (контролируемые) результаты освоения	
				Компетенции	Знания, умения, навыки
Задание 1. Линейные программы	Лабораторная работа	1	Активная	ПК-2	У1(ПК-2-1)
Задание 2. Ветвления и циклы	Лабораторная работа	1	Активная	ПК-2	У1(ПК-2-1)
Задание 3. Одномерные массивы и указатели	Лабораторная работа	1	Активная	ПК-2	У1(ПК-2-1)
Задание 4. Двумерные массивы	Лабораторная работа	1	Активная	ПК-2	У1(ПК-2-1)
Задание 5. Строки и файлы	Лабораторная работа	1	Активная	ПК-2	У1(ПК-2-1)
Задание 6. Структуры	Лабораторная работа	1	Активная	ПК-2	У1(ПК-2-1)
Задание 7. Функции	Лабораторная работа	0,5	Активная	ПК-2	У1(ПК-2-1)
Задание 8. Перегрузка и шаблоны функций	Лабораторная работа	0,5	Активная	ПК-2	У1(ПК-2-1)
Задание 9. Динамические структуры данных	Лабораторная работа	1	Активная	ПК-2	У1(ПК-2-1)
	Самостоятельная работа обучающихся Выполнение индивидуальных заданий лабораторных работ	76	Чтение основной и дополнительной литературы по темам раздела. Отработка	ПК-2	З1(ПК-2-1) У1(ПК-2-1) Н1(ПК-2-1)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)	Форма проведения	Планируемые (контролируемые) результаты освоения	
				Компетенции	Знания, умения, навыки
	рабочих работ раздела 1		примеров программ.		
Текущий контроль по разделу 1	Расчетно-графическая работа 1		Подготовка и защита отчета	ПК-2	З1(ПК-2-1) У1(ПК-2-1) Н1(ПК-2-1)
Промежуточная аттестация по дисциплине		4	Зачет с оценкой		
ИТОГО по разделу 1	Лекции	4	-	-	-
	Лабораторные работы	8	-	-	-
	Практические занятия		-	-	-
	Курсовое проектирование в аудитории		-	-	-
	Самостоятельная работа обучающихся	128	-	-	-
Раздел 2 Объектно-ориентированное программирование					
Тема 4. Классы Описание класса. Описание объектов. Конструкторы объектов.	Лекция	1	Активная	ПК-2	З1(ПК-2-2)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)	Форма проведения	Планируемые (контролируемые) результаты освоения	
				Компетенции	Знания, умения, навыки
Статические элементы класса. Дружественные функции. Деструкторы. Операции класса. Указатели на элементы класса. Вложенные и локальные классы.					
Тема 5. Наследование Формы наследования. Формы вложения.	Лекция	1	Традиционная	ПК-2	31(ПК-2-2)
Тема 6. Шаблоны классов. Обработка исключительных ситуаций Создание шаблона классов. Использование шаблона классов. Специализация шаблона классов. Принцип обработки исключений. Генерация исключений. Перехват исключений. Список исключений функции. Исключения в конструкторах и деструкторах. Стандартные исключения.	Лекция	1	Традиционная	ПК-2	31(ПК-2-2)
Тема 7. Стандартные потоки Потоковые классы. Перегрузка операций извлечения и вставки для пользовательских типов	Лекция	1	Традиционная	ПК-2	31(ПК-2-2)
Тема 8. Файловые и строковые по-	Лекция	1	Традици-	ПК-2	31(ПК-2-2)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)	Форма проведения	Планируемые (контролируемые) результаты освоения	
				Компетенции	Знания, умения, навыки
токи. Строки класса string Файловые потоки. Строковые потоки. Строки класса string.			онная		
Тема 9. Стандартная библиотека шаблонов Основные концепции стандартной библиотеки. Контейнеры. Итераторы. Алгоритмы.		1	Традиционная		31(ПК-2-2)
Задание 10. Классы	Лабораторная работа	2	Активная	ПК-2	У2(ПК-2-2)
Задание 11. Наследование	Лабораторная работа	2	Активная	ПК-2	У2(ПК-2-2)
Задание 12. Шаблоны классов и обработка исключительных ситуаций	Лабораторная работа	1	Активная	ПК-2	У2(ПК-2-2)
Задание 13. Стандартные потоки	Лабораторная работа	1	Активная	ПК-2	У2(ПК-2-2)
Задание 14. Файловые и строковые потоки. Строки класса string	Лабораторная работа	1	Активная	ПК-2	У2(ПК-2-2)
Задание 15. Стандартная библиотека шаблонов	Лабораторная работа	1	Активная	ПК-2	У2(ПК-2-2)
	Самостоятельная работа	157	Чтение основной и	ПК-2	32(ПК-2-2) У2(ПК-2-2)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)	Форма проведения	Планируемые (контролируемые) результаты освоения	
				Компетенции	Знания, умения, навыки
	та обучающихся Выполнение индивидуальных заданий лабораторных работ раздела 2		дополнительной литературы по темам раздела. Отработка примеров программ.		Н2(ПК-2-2)
Текущий контроль по разделу 2	Расчетно-графическая работа 2		Подготовка и защита отчета	ПК-2	32(ПК-2-2) У2(ПК-2-2) Н2(ПК-2-2)
Промежуточная аттестация по дисциплине		9	Экзамен		32(ПК-2-2) У2(ПК-2-2) Н2(ПК-2-2)
ИТОГО по разделу 2	Лекции	6	-	-	-
	Лабораторные работы	8	-	-	-
	Практические занятия		-	-	-
	Курсовое проектирование в аудитории		-	-	-
	Самостоятельная работа	157	-	-	-

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)	Форма проведения	Планируемые (контролируемые) результаты освоения	
	та обучающихся			Компетенции	Знания, умения, навыки
Раздел 3 Библиотеки классов и фреймворки					
Тема 10. Строки и регулярные выражения Строки. Регулярные выражения.	Лекция	1	Традиционная	ПК-2	31(ПК-2-3)
Тема 11. Контейнерные классы Последовательные контейнеры. Ассоциативные контейнеры.	Лекция	1	Традиционная	ПК-2	31(ПК-2-3)
Тема 12. Итераторы и функторы Итераторы. Функторы.	Лекция	0,5	Традиционная	ПК-2	31(ПК-2-3)
Тема 13. Алгоритмы Немодифицирующие операции с последовательностями. Модифицирующие операции с последовательностями. Алгоритмы связанные с сортировкой.	Лекция	0,5	Традиционная	ПК-2	31(ПК-2-3)
Тема 14. Средства для численных расчетов Обобщенные численные алгоритмы. Класс valarray. Комплексные числа. Генерация случайных последова-	Лекция	0,5	Традиционная	ПК-2	31(ПК-2-3)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)	Форма проведения	Планируемые (контролируемые) результаты освоения	
				Компетенции	Знания, умения, навыки
тельностью.					
Тема 15. Другие средства стандартной библиотеки Пары и кортежи. Интеллектуальные указатели.	Лекция	0,5	Традиционная	ПК-2	31(ПК-2-3)
Задание 16. Строки и регулярные выражения	Лабораторная работа	2	Активная	ПК-2	У1(ПК-2-3)
Задание 17. Контейнеры и алгоритмы	Лабораторная работа	2	Активная	ПК-2	У1(ПК-2-3)
Задание 18. Класс valarray	Лабораторная работа	2	Активная	ПК-2	У1(ПК-2-3)
	Самостоятельная работа обучающихся Выполнение индивидуальных заданий лабораторных работ раздела 3	94	Чтение основной и дополнительной литературы по темам раздела. Отработка примеров программ.	ПК-2	31(ПК-2-3) У1(ПК-2-3) Н1(ПК-2-3)
Текущий контроль по разделу 3	Курсовая работа		Подготовка и защита пояснительной	ПК-2	31(ПК-2-3) У1(ПК-2-3) Н1(ПК-2-3)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)	Форма проведения	Планируемые (контролируемые) результаты освоения	
				Компетенции	Знания, умения, навыки
			записки к курсовой работе		
Промежуточная аттестация по дисциплине		4	Зачет с оценкой		
ИТОГО по разделу 3	Лекции	4	-	-	-
	Лабораторные работы	6	-	-	-
	Практические занятия		-	-	-
	Курсовое проектирование в аудитории		-	-	-
	Самостоятельная работа обучающихся	94	-	-	-
ИТОГО по дисциплине	Лекции	14	-	-	-
	Лабораторные работы	22	-	-	-
	Практические занятия		-	-	-
	Курсовое проектирование		-	-	-

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)	Форма проведения	Планируемые (контролируемые) результаты освоения	
				Компетенции	Знания, умения, навыки
	вание в аудитории				
	Самостоятельная работа обучающихся	379	-	-	-
ИТОГО: общая трудоемкость дисциплины 432 часа, в том числе с использованием активных методов обучения 66 часов					

6 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Самостоятельная работа обучающихся, осваивающих дисциплину «Программирование на языке высокого уровня», состоит из следующих компонентов: изучение теоретических и практических разделов дисциплины; подготовка и оформление расчётно-графической работы, выполнение курсовой работы и оформление пояснительной записки.

Для успешного выполнения всех разделов самостоятельной работы учащимся рекомендуется использовать следующее учебно-методическое обеспечение:

Павловская, Т.А. С/С++. Процедурное и объектно-ориентированное программирование: учебник для вузов / Т.А. Павловская. — СПб.: Питер, 2015. — 496 с. : ил.

График выполнения самостоятельной работы по семестрам 2, 3, 4 представлен в таблицах 4.1, 4.2, 4.3.

Общие рекомендации по организации самостоятельной работы:

Время, которым располагает студент для выполнения учебного плана, складывается из двух составляющих: одна из них – это аудиторная работа в вузе по расписанию занятий, другая – внеаудиторная самостоятельная работа. Задания и материалы для самостоятельной работы выдаются во время учебных занятий по расписанию, на этих же занятиях преподаватель осуществляет контроль за самостоятельной работой, а также оказывает помощь студентам по правильной организации работы.

Чтобы выполнить весь объем самостоятельной работы, необходимо заниматься по 1 – 4 часа ежедневно. Начинать самостоятельные внеаудиторные занятия следует с первых же дней семестра. Первые дни семестра очень важны для того, чтобы включиться в работу, установить определенный порядок, равномерный ритм на весь семестр. Ритм в работе – это ежедневные самостоятельные занятия, желательно в одни и те же часы, при целесообразном чередовании занятий с перерывами для отдыха.

Начиная работу, не нужно стремиться делать вначале самую тяжелую ее часть, надо выбрать что-нибудь среднее по трудности, затем перейти к более трудной работе. И напоследок оставить легкую часть, требующую не столько больших интеллектуальных усилий, сколько определенных моторных действий (черчение, построение графиков и т.п.).

Следует правильно организовать свои занятия по времени: 50 минут – работа, 5-10 минут – перерыв; после 3 часов работы перерыв – 20-25 минут. Иначе нарастающее утомление повлечет неустойчивость внимания. Очень существенным фактором, влияющим на повышение умственной работоспособности, являются систематические занятия физической культурой. Организация активного отдыха предусматривает чередование умственной и физической деятельности, что полностью восстанавливает работоспособность.

Таблица 4.1 – Рекомендуемый график выполнения самостоятельной работы студентов при 17-недельном 2 семестре

Вид самостоятельной работы	Часов в неделю																	Итого по видам работ
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
Изучение теоретических и практических разделов дисциплины	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	2	2	64
Подготовка и оформление расчётно-графической работы	2	2	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	64
ИТОГО во 2 семестре	6	6	8	8	8	8	8	8	8	8	8	8	8	8	8	6	6	128

Таблица 4.2 – Рекомендуемый график выполнения самостоятельной работы студентов при 17-недельном 3 семестре

Вид самостоятельной работы	Часов в неделю																	Итого по видам работ
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
Изучение теоретических и практических разделов дисциплины	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	85
Подготовка и оформление расчётно-графической ра-	4	4	4	4	4	4	4	4	4	4	4	4	4	5	5	5	5	72

боты																		
ИТОГО в 3 семестре	9	9	9	9	9	9	9	9	9	9	9	9	9	10	10	10	10	157

Таблица 4.3 – Рекомендуемый график выполнения самостоятельной работы студентов при 17-недельном 4 семестре

Вид самостоя- тельной работы	Часов в неделю																	Итого по видам работ
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
Изучение теоре- тических и практических разделов дисци- плины	3	3	3	3	3	3	4	4	4	4	4	4	4	4	4	3	3	60
Выполнение курсовой работы и оформление пояснительной записки	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	3	3	36
ИТОГО в 4 семестре	5	5	5	5	5	5	6	6	6	6	6	6	6	6	6	6	6	96

7 Фонд оценочных средств для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине

Таблица 5 – Паспорт фонда оценочных средств

Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции (или ее части)	Наименование оценочного средства	Показатели оценки
Тема 1. Базовые средства языка	ПК-2-1	Задание 1. Линейные программы Задание 2. Ветвления и циклы Задание 3. Одномерные массивы и указатели Задание 4. Двумерные массивы Задание 5. Строки и файлы Задание 6. Структуры	Знает возможности существующей программно-технической архитектуры Умеет проводить оценку и обоснование рекомендуемых решений при разработке ПО
Тема 2. Модульное программирование	ПК-2-1	Задание 7. Функции Задание 8. Перегрузка и шаблоны функций	Знает возможности существующей программно-технической архитектуры Умеет проводить оценку и обоснование рекомендуемых решений при разработке ПО
Тема 3. Введение в организацию данных	ПК-2-1	Задание 9. Динамические структуры данных	Знает возможности существующей программно-технической архитектуры Умеет проводить оценку и обоснование рекомендуемых решений при разработке ПО
Тема 1 - 3	ПК-2-1	Расчетно-графическая работа 1	Владеет приемами разработки ПО на языках разного уровня
Тема 4. Классы	ПК-2-2	Задание 10. Классы	Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного

			обеспечения Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
Тема 5. Наследование	ПК-2-2	Задание 11. Наследование	Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
Тема 6. Шаблоны классов. Обработка исключительных ситуаций	ПК-2-2	Задание 12. Шаблоны классов и обработка исключительных ситуаций	Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
Тема 7. Стандартные потоки	ПК-2-2	Задание 13. Стандартные потоки	Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые

			при разработке программного обеспечения Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
Тема 8. Файловые и строковые потоки. Строки класса string	ПК-2-2	Задание 14. Файловые и строковые потоки. Строки класса string	Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
Тема 9. Стандартная библиотека шаблонов	ПК-2-2	Задание 15. Стандартная библиотека шаблонов	Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов
Тема 4 - 9	ПК-2-2	Расчетно-графическая работа 2	Владеет навыками использования библиотек программных моду-

			лей, шаблонов, классов объектов при разработке ПО
Тема 4 - 9	ПК-2-2	Экзаменационные билеты	Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения
Тема 10. Строки и регулярные выражения	ПК-2-3	Задание 16. Строки и регулярные выражения	Знает возможности существующей программно-технической архитектуры Умеет проводить оценку и обоснование рекомендуемых решений при разработке ПО
Тема 11. Контейнерные классы	ПК-2-3	Задание 17. Контейнеры и алгоритмы	Знает возможности существующей программно-технической архитектуры Умеет проводить оценку и обоснование рекомендуемых решений при разработке ПО
Тема 12. Итераторы и функторы	ПК-2-3	Задание 17. Контейнеры и алгоритмы	Знает возможности существующей программно-технической архитектуры Умеет проводить оценку и обоснование рекомендуемых решений при разработке ПО
Тема 13. Алгоритмы	ПК-2-3	Задание 17. Контейнеры и алгоритмы	Знает возможности существующей программно-технической архитектуры Умеет проводить оценку и обоснование рекомендуемых решений при

			разработке ПО
Тема 14. Средства для численных расчетов	ПК-2-3	Задание 18. Класс valarray	Знает возможности существующей программно-технической архитектуры Умеет проводить оценку и обоснование рекомендуемых решений при разработке ПО
Тема 10 - 14	ПК-2-3	Курсовая работа	Знает возможности существующей программно-технической архитектуры Умеет проводить оценку и обоснование рекомендуемых решений при разработке ПО Владеет навыками разработки ПО на базе классов

Промежуточная аттестация проводится в форме итоговой оценки во 2 и 4 семестрах, а в 3 семестре – в форме экзамена.

Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций, представлены в виде технологической карты дисциплины:

Таблица 6 – Технологическая карта

	Наименование оценочного средства	Сроки выполнения	Шкала оценива- ния	Критерии оценивания
2 семестр <i>Промежуточная аттестация в форме итоговой оценки</i>				
1	Задание 1 «Линейные программы»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
2	Задание 2 «Ветвления и циклы»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
3	Задание 3 «Одномерные массивы и указатели»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
4	Задание 4 «Двумерные массивы»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
5	Задание 5 «Строки и файлы»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
6	Задание 6 «Структуры»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
7	Задание 7 «Функции»	В течение семестра	8	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок

	Наименование оценочного средства	Сроки выполнения	Шкала оценива- ния	Критерии оценивания
				5 баллов – задание выполнено без недочетов и в срок
8	Задание 8 «Перегрузка и шаблоны функций»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
9	Задание 9 «Динамические структуры данных»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
ИТОГО:		-	45 баллов	-
Критерии оценки результатов обучения по дисциплине: 0 – 64 % от максимально возможной суммы баллов – «неудовлетворительно» (недостаточный уровень для текущей аттестации по дисциплине); 65 – 74 % от максимально возможной суммы баллов – «удовлетворительно» (пороговый (минимальный) уровень); 75 – 84 % от максимально возможной суммы баллов – «хорошо» (средний уровень); 85 – 100 % от максимально возможной суммы баллов – «отлично» (высокий (максимальный) уровень)				
3 семестр Промежуточная аттестация в форме экзамена				
1	Задание 10 «Классы»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
2	Задание 11 «Наследование»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
3	Задание 12 «Шаблоны классов и обработка исключительных ситуаций»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
4	Задание 13 «Стандартные потоки»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок

	Наименование оценочного средства	Сроки выполнения	Шкала оценива- ния	Критерии оценивания
5	Задание 14 «Файловые и строковые потоки. Стро- ки класса string»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
6	Задание 15 «Стандартная библиотека шаблонов»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
Текущая аттестация:		-	30	-
7	Экзамен	-	30	0 баллов – в ответах на вопросы допущено более двух ошибок 10 баллов – в ответах на вопросы допущено не более двух ошибок 20 баллов – в ответах на вопросы допущено не более одной ошибки 30 баллов – даны верные ответы на все вопросы билета
ИТОГО:		-	60 баллов	-
Критерии оценки результатов обучения по дисциплине: 0 – 64 % от максимально возможной суммы баллов – «неудовлетворительно» (недостаточный уровень для текущей аттестации по дисциплине); 65 – 74 % от максимально возможной суммы баллов – «удовлетворительно» (пороговый (минимальный) уровень); 75 – 84 % от максимально возможной суммы баллов – «хорошо» (средний уровень); 85 – 100 % от максимально возможной суммы баллов – «отлично» (высокий (максимальный) уровень)				
4 семестр <i>Промежуточная аттестация в форме итоговой оценки</i>				
1	Задание 16 «Строки и регулярные выражения»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
2	Задание 17 «Контейнеры и алгоритмы»	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
3	Задание 18	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок

	Наименование оценочного средства	Сроки выполнения	Шкала оценива- ния	Критерии оценивания
	«Класс valarray»			4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок
ИТОГО:		-	15 баллов	-
Критерии оценки результатов обучения по дисциплине: 0 – 64 % от максимально возможной суммы баллов – «неудовлетворительно» (недостаточный уровень для текущей аттестации по дисциплине); 65 – 74 % от максимально возможной суммы баллов – «удовлетворительно» (пороговый (минимальный) уровень); 75 – 84 % от максимально возможной суммы баллов – «хорошо» (средний уровень); 85 – 100 % от максимально возможной суммы баллов – «отлично» (высокий (максимальный) уровень)				
5	Курсовая работа	В течение семестра	5	0 баллов – задание не выполнено или выполнено не верно 3 балла – задание выполнено с недочетами и не в срок 4 балла – задание выполнено без недочетов и не в срок 5 баллов – задание выполнено без недочетов и в срок

Задания для текущего контроля

Задания для расчетно-графической работы 1 (семестр 2)

Задание 1. Написать программу расчета по двум формулам. Предварительно подготовить тестовые примеры.

1. Вариант 1

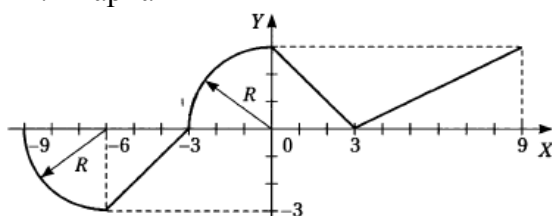
$$z1 = 2 \sin^2(3\pi - 2\alpha) \cos^2(5\pi + 2\alpha), \quad z2 = \frac{1}{4} - \frac{1}{4} \sin\left(\frac{5}{2}\pi - 8\alpha\right).$$

2. Вариант 2

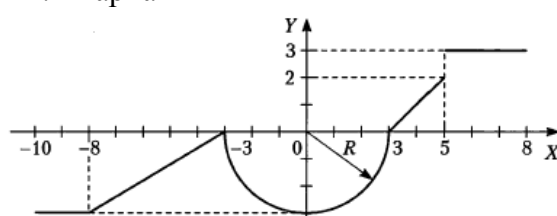
$$z1 = \cos \alpha + \sin \alpha + \cos 3\alpha + \sin 3\alpha, \quad z2 = 2\sqrt{2} \cos \alpha \cdot \sin\left(\frac{\pi}{4} - 2\alpha\right).$$

Задание 2. Разработать программу, вычисления значения функции, заданной графически. Результаты вывести на экран в виде таблицы на интервале от хнач до хкон с шагом dx. Значения хнач до хкон с шагом dx получить от пользователя.

1. Вариант 1



2. Вариант 2



Задание 3. Разработать программу обработки одномерного массива вещественных величин. Размерность массива задать именованной константой.

1. Вариант 1

1. Найти сумму отрицательных элементов массива.
2. Найти произведение элементов массива, расположенных между максимальным и минимальным элементами.
3. Упорядочить элементы массива по возрастанию.

2. Вариант 2

1. Найти сумму положительных элементов массива.
2. Найти произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами.
3. Упорядочить элементы массива по убыванию.

Задание 4. Разработать программу обработки двумерного массива.

1. Вариант 1

Дана целочисленная прямоугольная матрица. Определить:

- 1) количество строк, не содержащих ни одного нулевого элемента;
- 2) максимальное из чисел, встречающихся в заданной матрице более одного раза.

2. Вариант 2

Дана целочисленная прямоугольная матрица. Определить количество столбцов, не содержащих ни одного нулевого элемента.

Характеристикой строки целочисленной матрицы назовем сумму ее положительных четных элементов. Переставляя строки заданной матрицы, располагать их в соответствии с ростом характеристик.

Задание 5. Разработать программу обработки текстового файла с использованием строк в стиле C и символов.

1. Вариант 1

Написать программу, которая считывает из текстового файла три предложения и выводит их в обратном порядке.

2. Вариант 2

Написать программу, которая считывает текст из файла и выводит на экран только предложения, содержащие введенное с клавиатуры слово.

Задание 6. Описать структуру с заданным именем и полями. Разработать программу, обрабатывающую массив таких структур указанным способом.

1. Вариант 1

Описать структуру с именем STUDENT, содержащую следующие поля: фамилия и инициалы; номер группы; успеваемость (массив из пяти элементов). Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив, состоящий из пяти структур типа STUDENT; записи должны быть упорядочены по возрастанию номера группы;
- вывод на дисплей фамилий и номеров групп для всех студентов, включенных в массив, если средний балл студента больше 4.0;
- если таких студентов нет, вывести соответствующее сообщение.

2. Вариант 2

Описать структуру с именем STUDENT, содержащую следующие поля: фамилия и инициалы; номер группы; успеваемость (массив из пяти элементов). Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив, состоящий из пяти структур типа STUDENT; записи должны быть упорядочены по возрастанию среднего балла;
- вывод на дисплей фамилий и номеров групп для всех студентов, имеющих оценки и 5;
- если таких студентов нет, вывести соответствующее сообщение.

Задание 7. Выполнить задания 1, 2, 3, 4, 5, 6 оформив каждый пункт задания в виде функции. Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.

Задание 8. Выполнить задания 4, 5 оформив каждый пункт задания в виде шаблона функции. Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается. Привести примеры программ, использующих эти шаблоны для типов int, float и double.

Задание 9. Решить задачу, с применением наиболее подходящей динамической структуры данных. Разработать программу, демонстрирующую применение выбранной структуры данных для решения задач заданной предметной области.

1. Вариант 1

Составить программу, которая содержит динамическую информацию о наличии автобусов в автобусном парке. Сведения о каждом автобусе включают номер автобуса, фамилию, инициалы водителя и номер маршрута. Программа должна обеспечивать:

- начальное формирование данных обо всех автобусах в парке в виде списка;
- добавление и удаление автобусов;
- по запросу выдаются сведения об автобусах, находящихся в парке, или об автобусах, находящихся на маршруте.

2. Вариант 2

Составить программу, которая содержит текущую информацию о книгах в библиотеке. Сведения о книгах включают: номер УДК; фамилию и инициалы автора; название; год

издания; количество экземпляров данной книги в библиотеке.

Программа должна обеспечивать:

- начальное формирование данных о книгах в виде двоичного дерева;
- добавление и удаление книг;
- по запросу выдаются сведения о наличии книг в библиотеке, упорядоченные по годам издания.

Задания для расчетно-графической работы 2 (семестр 3)

Задание 10. Описать класс в нотации UML и разработать программу, демонстрирующую все возможности класса. Программа должна содержать меню, позволяющее выполнить проверку всех методов класса.

1. Вариант 1

Описать класс для представления обыкновенных целочисленных дробей. Обеспечить выполнение операций сложения, вычитания, умножения и деления дробей.

2. Вариант 2

Описать класс для представления комплексных чисел. Обеспечить выполнение операций сложения, вычитания и умножения комплексных чисел.

Задание 11. Написать программу, демонстрирующую работу с объектами двух типов, T1 и T2, для чего создать систему соответствующих классов. Каждый объект должен иметь идентификатор (в виде произвольной строки символов) и одно или несколько полей для хранения состояния (текущего значения) объекта.

Клиенту (функции main) должны быть доступны следующие основные операции (методы): создать объект, удалить объект, показать значение объекта и прочие дополнительные операции (зависят от варианта). Операции по созданию и удалению объектов инкапсулировать в классе Factory. Предусмотреть меню, позволяющее продемонстрировать заданные операции.

При необходимости в разрабатываемые классы добавляются дополнительные методы (например, конструктор копирования, операция присваивания и т.п.) для обеспечения надлежащего функционирования этих классов.

1. Вариант 1

Описать абстрактный класс Node для представления объектов некоторой файловой системы. На его основе создать производные классы Folder и File. Предусмотреть возможность вывода сведений об узле и построения карты всей файловой системы.

2. Вариант 2

Описать абстрактный класс Control для представления элемента управления. На его основе создать производные классы Panel и Button. Предусмотреть возможность вывода сведений об элементах управления и построения дерева принадлежности элементов управления.

Задание 12. Требуется создать шаблон некоторого целевого класса A. В каждом варианте уточняются требования к реализации — указанием на применение некоторого серверного класса B. Это означает, что объект класса B используется как элемент класса A. В качестве серверного класса может быть указан либо класс, созданный программистом в рамках того же задания, либо класс стандартной библиотеки.

1. Вариант 1

Создать шаблон класса List. Использовать его при решении задачи варианта 1 задания 9.

2. Вариант 2

Создать шаблон класса Tree. Использовать его при решении задачи варианта 2 задания 9.

Задание 13. Написать программу и продемонстрировать использование всех разработанных элементов класса, согласно заданию.

1. Вариант 1

Определить класс с именем STUDENT, содержащий следующие поля: фамилия и инициалы; номер группы; успеваемость (массив из пяти элементов). Определить методы доступа к этим полям и перегруженные операции извлечения и вставки для объектов типа STUDENT.

2. Вариант 2

Определить класс с именем TRAIN, содержащий следующие поля: название пункта назначения; номер поезда; время отправления. Определить методы доступа к этим полям и перегруженные операции извлечения и вставки для объектов типа TRAIN.

Задание 14. Выполнить задание 5, используя потоковые классы и класс string.

Задание 15. Выполнить задачи задания используя для этого классы из библиотеки STL.

1. Вариант 1

Написать программу для моделирования Т-образного сортировочного узла на железной дороге с использованием контейнерного класса stack из STL.

Программа должна разделять на два направления состав, состоящий из вагонов двух типов (на каждое направление формируется состав из вагонов одного типа).

Предусмотреть возможность ввода исходных данных с клавиатуры и из файла.

2. Вариант 2

Написать программу «Моя записная книжка». Предусмотреть возможность работы с произвольным числом записей, поиска записи по какому-либо признаку (например, по фамилии, дате рождения или номеру телефона), добавления и удаления записей, сортировки по разным полям.

Хранение данных организовать с помощью класса map или multimap.

Задания лабораторных работ (семестр 4)

Задание 16. Выполнить задание 5 с использованием класса string и регулярных выражений.

Задание 17. Выполнить задание 3 с использованием подходящих контейнерных классов и алгоритмов библиотеки. Содержимое контейнеров (не менее миллиона элементов) формировать с помощью генераторов случайных чисел. Замерить время обработки с помощью средств стандартной библиотеки (заголовочный файл <chrono>).

Сравнить полученный результат с результатом работы функций, полученных при выполнении задания 7.

Задание 18. Выполнить задания 3 и 4 с использованием класса valarray, срезов и генераторов случайных чисел.

Задания для курсовой работы (семестр 4)

Задание 1. Написать программу расчета по двум формулам. Предварительно подготовить тестовые примеры.

3. Вариант 1

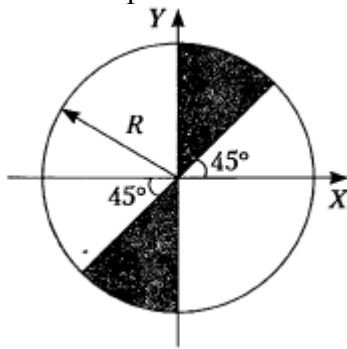
$$z1 = 2 \sin^2(3\pi - 2\alpha) \cos^2(5\pi + 2\alpha), \quad z2 = \frac{1}{4} - \frac{1}{4} \sin\left(\frac{5}{2}\pi - 8\alpha\right).$$

4. Вариант 2

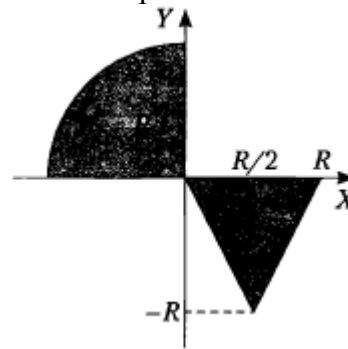
$$z1 = \cos \alpha + \sin \alpha + \cos 3\alpha + \sin 3\alpha, \quad z2 = 2\sqrt{2} \cos \alpha \cdot \sin\left(\frac{\pi}{4} - 2\alpha\right).$$

Задание 2. Написать программу, определяющую попадание точки с определенными координатами в заданную плоскую область. Значения координат вводить с клавиатуры. Результат работы программы вывести в виде текстового сообщения

1. Вариант 1

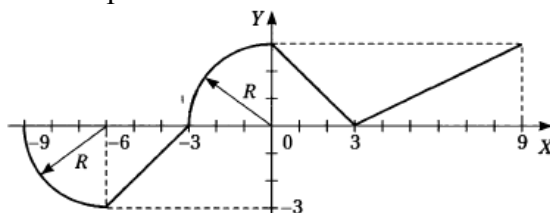


2. Вариант 2

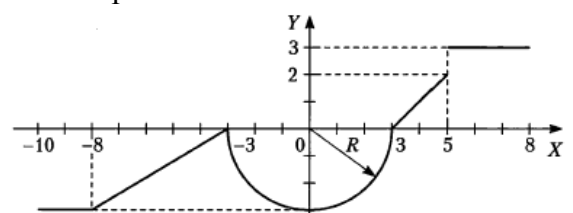


Задание 3. Разработать программу, вычисления значения функции, заданной графически. Результаты вывести на экран в виде таблицы на интервале от $x_{нач}$ до $x_{кон}$ с шагом dx . Значения $x_{нач}$ до $x_{кон}$ с шагом dx получить от пользователя.

3. Вариант 1



4. Вариант 2



Задание 4. Каждый разрабатываемый класс должен содержать следующие элементы: скрытые поля, конструкторы с параметрами и без параметров, методы, свойства. Методы и свойства должны обеспечивать непротиворечивый, полный, минимальный и удобный интерфейс класса. При возникновении ошибок должны выбрасываться исключения. В программе должна выполняться проверка всех разработанных элементов класса.

3. Вариант 1

Описать класс, реализующий десятичный счетчик, который может увеличивать или уменьшать свое значение на единицу в заданном диапазоне. Предусмотреть инициализацию счетчика значениями по умолчанию и произвольными значениями. Счетчик имеет два метода: увеличения и уменьшения, и свойство, позволяющее получить его текущее состояние. При выходе за границы диапазона выбрасываются исключения. Продемонстрировать применение всех разработанных элементов класса.

4. Вариант 2

Описать класс, реализующий шестнадцатеричный счетчик, который может увели-

чивать или уменьшать свое значение на единицу в заданном диапазоне. Предусмотреть инициализацию счетчика значениями по умолчанию и произвольными значениями. Счетчик имеет два метода: увеличения и уменьшения, и свойство, позволяющее получить его текущее состояние. При выходе за границы диапазона выбрасываются исключения. Продемонстрировать применение всех разработанных элементов класса.

Задание 5. Разработать программу обработки одномерного массива вещественных величин.

1. Вариант 1
 1. Найти сумму отрицательных элементов массива.
 2. Найти произведение элементов массива, расположенных между максимальным и минимальным элементами.
 3. Упорядочить элементы массива по возрастанию.
2. Вариант 2
 1. Найти сумму положительных элементов массива.
 2. Найти произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами.
 3. Упорядочить элементы массива по убыванию.

Задание 6. Разработать программу обработки двумерного массива.

3. Вариант 1

Дана целочисленная прямоугольная матрица. Определить:

 - 1) количество строк, не содержащих ни одного нулевого элемента;
 - 2) максимальное из чисел, встречающихся в заданной матрице более одного раза.
4. Вариант 2

Дана целочисленная прямоугольная матрица. Определить количество столбцов, не содержащих ни одного нулевого элемента.

Характеристикой строки целочисленной матрицы назовем сумму ее положительных четных элементов. Переставляя строки заданной матрицы, располагать их в соответствии с ростом характеристик.

Задание 7. Разработать программу обработки текстового файла.

3. Вариант 1

Написать программу, которая считывает из текстового файла три предложения и выводит их в обратном порядке.
4. Вариант 2

Написать программу, которая считывает текст из файла и выводит на экран только предложения, содержащие введенное с клавиатуры слово.

Задание 8. Каждый разрабатываемый класс должен, как правило, содержать следующие элементы: скрытые поля, конструкторы с параметрами и без параметров, методы; свойства, индексы; перегруженные операции. Функциональные элементы класса должны обеспечивать непротиворечивый, полный, минимальный и удобный интерфейс класса. При возникновении ошибок должны выбрасываться исключения. В программе должна выполняться проверка всех разработанных элементов класса.

3. Вариант 1

Описать класс для работы с одномерным массивом целых чисел (вектором). Обеспечить следующие возможности:

 - задание произвольных целых границ индексов при создании объекта;
 - обращение к отдельному элементу массива с контролем выхода за пределы массива;

- выполнение операций поэлементного сложения и вычитания массивов с одинаковыми границами индексов;
- выполнение операций умножения и деления всех элементов массива на скаляр;
- вывод на экран элемента массива по заданному индексу и всего массива.

Написать программу, демонстрирующую все разработанные элементы класса.

4. Вариант 2

Описать класс для работы с одномерным массивом строк фиксированной длины. Обеспечить следующие возможности:

- задание произвольных целых границ индексов при создании объекта;
- обращение к отдельной строке массива по индексу с контролем выхода за пределы массива;
- выполнение операций поэлементного сцепления двух массивов с образованием нового массива;
- выполнение операций слияния двух массивов с исключением повторяющихся элементов;
- вывод на экран элемента массива по заданному индексу и всего массива.

Написать программу, демонстрирующую все разработанные элементы класса.

Задание 9. В программах требуется описать базовый класс (возможно, абстрактный), в котором с помощью виртуальных или абстрактных методов и свойств задается интерфейс для производных классов. Целью лабораторной работы является максимальное использование наследования, даже если для конкретной задачи оно не дает выигрыша в объеме программы. Во всех классах следует переопределить метод Equals, чтобы обеспечить сравнение значений, а не ссылок.

Функция Main должна содержать массив из элементов базового класса, заполненный ссылками на производные классы. В этой функции должно демонстрироваться использование всех разработанных элементов классов.

1. Вариант 1

Создать класс Point (точка). На его основе создать классы ColoredPoint и Line (линия). На основе класса Line создать классы ColoredLine и PolyLine (многоугольник). В классах описать следующие элементы:

- конструкторы с параметрами и конструкторы по умолчанию;
- свойства для установки и получения значений всех координат, а также для изменения цвета и получения текущего цвета;
- для линий—методы изменения угла поворота линий относительно первой точки;
- для многоугольника—метод масштабирования.

2. Вариант 2

Создать абстрактный класс Vehicle (транспортное средство). На его основе реализовать классы Plane (самолет), Car (автомобиль) и Ship (корабль). Классы должны иметь возможность задавать и получать координаты и параметры средств передвижения (цена, скорость, год выпуска и т. п.) с помощью свойств. Для самолета должна быть определена высота, для самолета и корабля – количество пассажиров, для корабля – порт приписки. Динамические характеристики задать с помощью методов.

Задание 10. Описать структуру с заданным именем и полями. Разработать программу, обрабатывающую массив таких структур указанным способом.

1. Вариант 1

Описать структуру с именем STUDENT, содержащую следующие поля: фамилия и инициалы; номер группы; успеваемость (массив из пяти элементов). Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив, состоящий из пяти структур типа STUDENT; записи должны быть упорядочены по возрастанию номера группы;
- вывод на дисплей фамилий и номеров групп для всех студентов, включенных в массив, если средний балл студента больше 4.0;
- если таких студентов нет, вывести соответствующее сообщение.

2. Вариант 2

Описать структуру с именем STUDENT, содержащую следующие поля: фамилия и инициалы; номер группы; успеваемость (массив из пяти элементов). Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив, состоящий из пяти структур типа STUDENT; записи должны быть упорядочены по возрастанию среднего балла;
- вывод на дисплей фамилий и номеров групп для всех студентов, имеющих оценки и 5;
- если таких студентов нет, вывести соответствующее сообщение.

Задание 11. Выполнить задания 9, используя для хранения экземпляров разработанных классов стандартные параметризованные коллекции. Во всех классах реализовать интерфейс IComparable и перегрузить операции отношения для реализации значимой семантики сравнения объектов по какому-либо полю на усмотрение студента.

Задание 12. Написать Windows-приложение, заголовок главного окна которого содержит Ф. И. О., группу студента и номер варианта. В программе должна быть предусмотрена обработка исключений, возникающих из-за ошибочного ввода.

1. Вариант 1

Создать меню с командами Input, Calc и Exit.

При выборе команды Input открывается диалоговое окно, содержащее:

- три поля типа Text Box для ввода длин трех сторон треугольника;
- группу из двух флажков (Периметр и Площадь) типа CheckBox;
- кнопку типа Button.

Обеспечить возможность:

- ввода длин трех сторон треугольника;
- выбора режима с помощью флажков: подсчет периметра и/или площади треугольника

При выборе команды Calc открывается диалоговое окно с результатами. При выборе команды Exit приложение завершается.

2. Вариант 2

Создать меню с командами Size, Color, Paint, Quit.

Команда Paint недоступна. При выборе команды Quit приложение завершается.

При выборе команды Size открывается диалоговое окно, содержащее:

- два поля типа Text Box для ввода длин сторон прямоугольника;
- группу из трех флажков (Red, Green, Blue) типа CheckBox;
- кнопку типа Button.

Обеспечить возможность:

- ввода длин сторон прямоугольника в пикселях в поля ввода;
- выбора его цвета с помощью флажков.

После задания параметров команда Paint становится доступной.

При выборе команды Paint в главном окне приложения выводится прямоугольник заданного размера и сочетания цветов или выдается сообщение, если введенные размеры превышают размер окна.

Вопросы экзамена (семестр 3)

1. Описание класса
2. Описание объектов
3. Указатель `this`
4. Конструкторы
5. Конструктор копирования
6. Статические элементы класса
7. Статические поля
8. Статические методы
9. Дружественные функции и классы
10. Дружественная функция
11. Дружественный класс
12. Деструкторы
13. Перегрузка операций
14. Перегрузка унарных операций
15. Перегрузка бинарных операций
16. Перегрузка операции присваивания
17. Перегрузка операций `new` и `delete`
18. Перегрузка операции приведения типа
19. Перегрузка операции вызова функции
20. Перегрузка операции индексирования
21. Указатели на элементы классов
22. Рекомендации по составу класса
23. Ключи доступа
24. Простое наследование
25. Виртуальные методы
26. Механизм позднего связывания
27. Абстрактные классы
28. Множественное наследование
29. Отличия структур и объединений от классов
30. Создание шаблонов классов
31. Использование шаблонов классов
32. Специализация шаблонов классов
33. Достоинства и недостатки шаблонов
34. Общий механизм обработки исключений
35. Синтаксис исключений
36. Перехват исключений
37. Список исключений функции
38. Исключения в конструкторах и деструкторах
39. Иерархии исключений
40. Операция приведения типов в стиле C
41. Операция `const_cast`
42. Операция `dynamic_cast`
43. Повышающее преобразование
44. Понижающее преобразование
45. Преобразование ссылок
46. Перекрестное преобразование
47. Операция `static_cast`
48. Операция `reinterpret_cast`
49. Динамическое определение типа
50. Стандартные потоки
51. Форматирование данных
52. Флаги и форматирующие методы
53. Манипуляторы
54. Методы обмена с потоками
55. Ошибки потоков
56. Файловые потоки
57. Строковые потоки
58. Потоки и типы, определенные пользователем
59. Конструкторы и присваивание строк
60. Операции
61. Функции
62. Присваивание и добавление частей строк
63. Преобразования строк
64. Поиск подстрок
65. Сравнение частей строк
66. Получение характеристик строк
67. Последовательные контейнеры
68. Векторы (`vector`)
69. Двусторонние очереди (`deque`)
70. Списки (`list`)
71. Стеки (`stack`)
72. Очереди (`queue`)
73. Очереди с приоритетами (`priority_queue`)
74. Ассоциативные контейнеры
75. Словари (`map`)
76. Словари с дубликатами (`multimap`)
77. Множества (`set`)
78. Множества с дубликатами (`multiset`)
79. Битовые множества (`bitset`)
80. Итераторы и функциональные объекты

Примерные варианты экзаменационных билетов (семестр 3)

Билет №1

1. Описание класса
2. Простое наследование
3. Последовательные контейнеры

Билет №2

1. Дружественные функции и классы
2. Создание шаблонов классов
3. Ассоциативные контейнеры

8 Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

8.2 Основная литература

1 Немцова, Т. И. Программирование на языке C++ [Электронный ресурс]: учеб. пособие / Т.И. Немцова, С.Ю. Голова, А.И. Терентьев; Под ред. Л.Г. Гагариной. - М.: ИД ФОРУМ: ИНФРА-М, 2012. - 512 с. // ZNANIUM.COM : электронно-библиотечная система. – Режим доступа: <http://znanium.com/catalog.php#>, ограниченный. – Загл. с экрана.

8.2 Дополнительная литература

1 Павловская, Т. А. C/C++. Программирование на языке высокого уровня: учебник для вузов / Т. А. Павловская. – СПб. : Питер, 2010; 2003; 2001. – 460 с.

2 Павловская, Т.А. С #: Программирование на языке высокого уровня: Учебник для вузов / Т. А. Павловская. - СПб.: Питер, 2010; 2007. - 432с.

9 Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее – сеть «Интернет»), необходимых для освоения дисциплины (модуля)

C++ Standard Library Reference [Электронный ресурс] / Colin Robertson, Mike Blome, Gordon Hogenson, Saisang Cai. Дата обновления: 04.11.2016. – Режим доступа: <https://docs.microsoft.com/ru-ru/cpp/standard-library/cpp-standard-library-reference>, свободный. – Загл. с экрана. Яз. англ.

10 Методические указания для обучающихся по освоению дисциплины (модуля)

Обучение дисциплине «Программирование на языке высокого уровня» предполагает изучение курса на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проводятся в форме лекций и лабораторных работ. Самостоятельная работа включает:

- изучение теоретических и практических разделов дисциплины;
- подготовку и оформление расчётно-графической работы;
- выполнение курсовой работы и оформление пояснительной записки.

Таблица 7 – Методические указания к отдельным видам деятельности

Вид учебного занятия	Организация деятельности студента
Лекция	Рисование карт мышления (MindMap). Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения. Выделять ключевые слова, формулы, отмечать на полях уточняющие вопросы по теме занятия.
Лабораторная работа	Работа с картой мышления (конспектом лекций), изучение разделов основной литературы по теме занятия, работа с текстом, освоение электронных материалов по дисциплине, отработка решения задач по приведенным примерам
Самостоятельная работа	Для более глубокого изучения разделов дисциплины предусмотрены отдельные виды самостоятельной работы: изучение теоретических и практических разделов дисциплины; подготовку и оформление расчётно-графической работы; выполнение курсовой работы и оформление пояснительной записки. Более подробно структура и содержание самостоятельной работы описаны в разделе 6.

В качестве опорного конспекта лекций используется учебник:

Павловская, Т.А. С/С++. Процедурное и объектно-ориентированное программирование: учебник для вузов / Т.А. Павловская. — СПб.: Питер, 2015. — 496 с. : ил.

Составление отчетов к РГР и пояснительной записки к КР

Отчеты к заданиям выполняются в соответствии с требованиями РД 013-2016 «Текстовые студенческие работы. Правила оформления» и подшивается в папку (типа «скоросшиватель»). В общем случае структурными элементами РГР/КР являются (СТО У.003-2017):

- титульный лист;
- задание на РГР/КР;
- содержание;
- основная часть;
- оценка результатов проектирования;
- заключение;
- список использованных источников;
- приложения.

Во введении дается краткое обоснование обращения к выбранной теме, раскрывается актуальность, новизна, теоретическая и практическая значимость работы и приводится программа исследования. Программа исследования представляет собой изложение концепции исследования и включает следующие элементы: объект, предмет, цель, задачи.

Объектом принято считать то, на что направлена познавательная деятельность обучающегося. Объект исследования является более широким понятием, чем предмет.

Предметом могут быть существенные свойства или отношения объекта исследования, познание которых важно для решения теоретических или практических проблем. Предмет исследования определяет границы изучения объекта в конкретной РГР/КР.

Цель – конечный результат исследования, то, ради чего оно проводится.

Задачи исследования – промежуточные этапы, достижение которых необходимо для реализации поставленной цели. Объем введения: не более 2-3 страниц текста.

Каждое выполненное индивидуальное задание оформляется отдельным разделом основной части отчета. Каждый раздел включает теоретические сведения по теме раздела с обязательным указанием ссылок на источники информации, в случае её заимствования. Обязательными подразделами каждого отчета являются «Текст программы», «Описание программы», «Программа и методика испытаний». Требования к содержанию подразделов приведены в таблице:

Вид программного документа	Содержание программного документа
Текст программы	Запись программы с необходимыми комментариями
Описание программы	Сведения о логической структуре и функционировании программы
Программа и методика испытаний	Требования, подлежащие проверке при испытании программы, а также порядок и методы их контроля

В заключении приводятся общие выводы и предложения, а также краткое описание проделанной работы; даются практические рекомендации. (1,5 - 2 страницы).

Список литературы состоит из нормативно-правовых актов, учебников и учебных пособий, научных статей, использованных в ходе выполнения индивидуального задания.

Приложения помещают после списка литературы в порядке их отсылки или обращения к ним в тексте. В качестве приложений рекомендуется предоставлять копии документов, бланков договоров, организационно-распорядительных документов, аналитических таблиц, иных документов, иллюстрирующих содержание основной части.

Отчет предварительно оценивается и допускается к защите после проверки его соответствия требованиям, предъявляемым данными методическими указаниями. Защита отчетов организуется в форме собеседования.

11 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)

С целью повышения качества ведения образовательной деятельности в университете создана электронная информационно-образовательная среда. Она подразумевает организацию взаимодействия между обучающимися и преподавателями через систему личных кабинетов студентов, расположенных на официальном сайте университета в информационно телекоммуникационной сети «Интернет» по адресу <https://student.knastu.ru>. Созданная информационно-образовательная среда позволяет осуществлять взаимодействие между участниками образовательного процесса посредством организации дистанционного консультирования по вопросам выполнения практических заданий.

Рекомендуемая среда разработки **Visual Studio Community 2017** или более поздняя версия. Вот как Visual Studio Community можно использовать в организациях:

Visual Studio Community может использовать неограниченное число пользователей в организации в следующих случаях: в учебных аудиториях, для научных исследований или участия в проектах с открытым кодом.

Для всех прочих сценариев использования:

В некорпоративных организациях Visual Studio Community могут использовать до 5 пользователей. В корпоративных организациях (в которых используется более 250 ПК или годовой доход которых превышает 1 млн долларов США) использование запрещено, за исключением случаев, перечисленных выше (открытый код, научные исследования и учебные аудитории)¹.

¹ <https://www.visualstudio.com/ru/vs/community/>

12 Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Для реализации программы дисциплины «Программирование на языке высокого уровня» используется материально-техническое обеспечение, перечисленное в таблице 7.

Таблица 7 – Материально-техническое обеспечение дисциплины

Аудитория	Наименование аудитории (лаборатории)	Используемое оборудование	Назначение оборудования
228-1	Лаборатория ГИС технологий	9 персональных ЭВМ с процессором Core(TM) i5-3240 CPU @ 3.5 GHz; 1 экран с проектором	Проведение лекционных и лабораторных занятий
321-3	Лаборатория мультимедийных технологий	10 персональных ЭВМ с процессором Core(TM) i5-4690 CPU @ 3.5 GHz; 1 экран с проектором	Проведение лекционных и лабораторных занятий

Приложение А

Задания для организации «входного» контроля знаний учащихся.

1. Что из указанного ниже является элементами программ:
 - а) операторы;
 - б) компиляторы;
 - в) переменные;
 - г) производные.
2. Зачем нужен тип данных
 - а) для определения допустимых операций;
 - б) для обеспечения целостности данных;
 - в) для создания точек остановки программ;
 - г) для указания переменной типа ее содержимого.
3. Укажите, что является результатом выполнения оператора присваивания:
 - а) изменение значения переменной на заданное значение;
 - б) изменение значения константы на вычисленное значение;
 - в) передача управления подпрограмме;
 - г) проверка на равенство двух величин.
4. Для чего предназначен оператор цикла:
 - а) для передачи управления подпрограмме;
 - б) для продолжения работы программы после остановки;
 - в) для организации многократного выполнения группы операторов;
 - г) для выбора одного из нескольких вариантов дальнейших действий.
5. Что такое массив
 - а) именованный набор переменных, имеющих различные типы данных, и располагающихся в одной области памяти;
 - б) переменный набор имен, имеющий один тип и разное место в памяти;
 - в) набор переменных имен и функций, которые располагаются в одной области памяти;
 - г) именованный набор переменных, имеющий один тип данных, и располагающихся в одной области памяти.
6. Что такое функция
 - а) некоторая часть программы имеющая собственное имя, которая может быть вызвана необходимое количество раз;
 - б) некоторая часть программы содержащая вредоносный код, и блокирующая определенные действия системы;
 - в) часть программы, в которой хранятся глобальные переменные;
 - г) некоторое действие, не имеющее возвращаемого результата.
7. В чем состоит основное назначение модульного программирования:
 - а) сокращение размеров программ;
 - б) упрощение решения задачи за счет разбиения её на подзадачи;
 - в) ускорение выполнения вычислений;
 - г) создание библиотек полезных подпрограмм.

Лист регистрации изменений к РПД

№ п/п	Содержание изменения/основание	Кол-во стр. РПД	Подпись автора РПД
1	Изменение листа подписей в связи со сменой декана ФКТ /пр.№ 271-ЛС «к» от 29.12.2016	1	
2	Изменение КУГ/пр. № 326-О «а» от 04.09.2017	7	
3	Изменение титульного листа в связи с переименованием вуза/пр. №997-О от 03.11.2017	1	
4	Актуализация литературы/ 28.11.2017	2	