

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ
ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ



УТВЕРЖДАЮ
Декан факультета

Л. Р. Фионова

(Подпись)

(Фамилия, инициалы)

2015 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.1.24 Программирование для ЭВМ

Направление подготовки 01.03.04 — «Прикладная математика»

Профиль подготовки «Математическое моделирование в экономике и технике»

Квалификация (степень) выпускника – *бакалавр*

Форма обучения очная

1. Цели освоения дисциплины

Целями освоения дисциплины «Программирование для ЭВМ» являются:

- изучение основ алгоритмизации инженерных и математических задач;
- формирование у студентов глубоких теоретических и практических знаний основ программирования;
- формирование практических навыков работы в средах разработки Microsoft Visual Studio (язык C/C++).
- выработка умения студентами самостоятельно расширять знания основ программирования.

2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина «Программирование для ЭВМ» относится к базовой части профессионального блока Б.1. и является одной из дисциплин, формирующих профессиональные знания и навыки, характерные для бакалавра по направлению подготовки 01.03.04 «Прикладная математика».

Для успешного освоения дисциплины студенты должны иметь знания, полученные в рамках курса основ информатики и вычислительной техники, а также курса элементарной и высшей математики средней школы. Кроме того, необходимы элементарные навыки работы в операционной системе Windows.

Дисциплина «Программирование для ЭВМ» служит основой для дальнейшего изучения таких дисциплин как:

- «Теория графов и математическая логика», «Методы оптимизации», «Математическое моделирование», «Численные методы», «Программные и аппаратные средства информатики», «Компьютерная графика», «Операционные системы и сети ЭВМ», «Базы данных» (базовая часть Б.1);
- «Архитектура ЭВМ», «Дискретная математика», «Вычислительная математика», «Проектирование программного обеспечения», «Программирование для Интернет» (вариативная часть Б.1).

3. Компетенции обучающегося, формируемые в результате освоения дисциплины «Программирование для ЭВМ»

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО по данному направлению:

Коды компетенции	Наименование компетенции	Структурные элементы компетенции (в результате освоения дисциплины обучающийся должен знать, уметь, владеть)
1	2	3
ОПК-2	способность использовать современные математические методы и современные прикладные программные средства и осваивать современные технологии программирования	<p>Знать:</p> <ul style="list-style-type: none"> – особенности математических вычислений, реализуемых на ЭВМ; – методы исследования основных инженерных и научных задач проектирования и разработки объектов профессиональной деятельности; – основные структуры данных и методы их обработки; <p>Уметь:</p> <ul style="list-style-type: none"> – проводить разработку и анализ алгоритмов; – программировать алгоритм, используя средства языка высокого уровня; <p>Владеть:</p> <ul style="list-style-type: none"> – способностью формализовать прикладную задачу, выбрать для нее подходящие структуры данных и алгоритмы обработки;
ПК-1	способность использовать стандартные пакеты прикладных программ для решения практических задач на электронных вычислительных машинах, отлаживать, тестировать прикладное программное обеспечение	<p>Знать:</p> <ul style="list-style-type: none"> – конкретный язык программирования; – типы вычислительных ошибок; – набор функций стандартной библиотеки; <p>Уметь:</p> <ul style="list-style-type: none"> – составлять алгоритмы с учетом специфики машинных вычислений; <p>Владеть:</p> <ul style="list-style-type: none"> – способностью формализовать прикладную задачу, выбрать для нее подходящие структуры данных и алгоритмы обработки; – разрабатывать программу для ЭВМ, проводить ее отладку и тестирование, оформлять документацию на программу.

1	2	3
ПК-3	<p>способность и готовность демонстрировать знания современных языков программирования, операционных систем, офисных приложений, информационно-телекоммуникационной сети «Интернет», способов и механизмов управления данными, принципов организации, состава и схемы работы операционных систем</p>	<p>Знать:</p> <ul style="list-style-type: none"> – архитектуру персонального компьютера; – общие принципы построения вычислительных алгоритмов; – компьютерную систему чисел с плавающей точкой; – методы исследования основных инженерных и научных задач проектирования и разработки объектов профессиональной деятельности; <p>Уметь:</p> <ul style="list-style-type: none"> – проводить разработку и анализ алгоритмов; – программировать алгоритм, используя средства языка высокого уровня; <p>Владеть:</p> <ul style="list-style-type: none"> – разрабатывать программу для ЭВМ, проводить ее отладку и тестирование, оформлять документацию на программу.

4. Структура и содержание дисциплины «Программирование для ЭВМ»

4.1. Структура дисциплины

Общая трудоемкость дисциплины составляет **8** зачетных единиц, **288** часа.

Общая трудоемкость дисциплины за первый семестр составляет **3** зачетных единиц, **108** часов.

№ п/п	Наименование разделов и тем дисциплины (модуля)	Семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)									Формы текущего контроля успеваемости (по неделям семестра)							
				Аудиторная работа				Самостоятельная работа					Собеседование	Коллоквиум	Проверка тестов	Проверка контролльн. работ	Проверка реферата	Проверка эссе и иных творческих работ	курсовая работа (проект)	др.
				Всего	Лекция	Практические занятия	Лабораторные занятия	Всего	Подготовка к аудиторным занятиям	Реферат, эссе и др.	Курсовая работа (проект)	Подготовка к зачету								
1.	Раздел 1. Основы архитектуры семейства 80x86.	1	1–2	4	2		2	2	2											
1.1	Тема 1.1. Архитектура ЭВМ по фон Нейману.	1	1–2	4	2		2	2	2				5							
2.	Раздел 2. Этапы разработки программы. Структура алгоритма. Языки проектирования алгоритмов	1	1–4	12	2	4	6	6	6											
2.1.	Тема 2.1. Этапы решения задач на ЭВМ.	1	1–2	4		2	2	2	2				5							
2.2.	Тема 2.2. Алгоритмизация	1	3–4	4	2		2	2	2				5							
2.3.	Тема 2.3. Языки проектирования алгоритмов	1	3–4	4		2	2	2	2				5							
3.	Раздел 3. Основы программирования на языке Си	1	5–17	56	14	14	28	28	24			4								

№ п/п	Наименование разделов и тем дисциплины (модуля)	Семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)								Формы текущего контроля успеваемости (по неделям семестра)							
				Аудиторная работа				Самостоятельная работа				Собеседование	Коллоквиум	Проверка тестов	Проверка контрольн. работ	Проверка реферата	Проверка эссе и иных твор- ческих работ	курсовая работа (проект)	др.
				Всего	Лекция	Практические занятия	Лабораторные занятия	Всего	Подготовка к аудиторным занятиям	Реферат, эссе и др.	Курсовая работа (проект)								
3.1.	Тема 3.1. Процесс построения исполняемой программы. Структура программа (директивы препроцессора, библиотеки функций, функция main).	1	5–6	4	2		2	2	2				10						
3.2.	Тема 3.2. Представление чисел в компьютере. Форматы данных для представления целых чисел и плавающих чисел: одинарный, двойной, расширенный. Сопроцессор.	1	5–6	4		2	2	2	2				10						
3.3.	Тема 3.3. Переменные и константы. Область действия и область видимости переменной. Базовые типы данных (допустимые диапазоны значений переменных различных типов).	1	7–8	4	2		2	2	2				10						
3.4.	Тема 3.4. Управляющие структуры языка C/C++: следование; выбор; повторение.	1	7–8	4		2	2	2,5	2			0,5	10						
3.5.	Тема 3.5. Ввод/вывод в C/C++. printf(), scanf(), cin, cout. Пространство имен.	1	9–10	4	2		2	2,5	2			0,5	10						

№ п/п	Наименование разделов и тем дисциплины (модуля)	Семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)								Формы текущего контроля успеваемости (по неделям семестра)								
				Аудиторная работа				Самостоятельная работа				Собеседование	Коллоквиум	Проверка тестов	Проверка контрольн. работ	Проверка реферата	Проверка эссе и иных твор- ческих работ	курсовая работа (проект)	др.	
				Всего	Лекция	Практические занятия	Лабораторные занятия	Всего	Подготовка к аудиторным занятиям	Реферат, эссе и др.	Курсовая работа (проект)									Подготовка к экзамену
3.6.	Тема 3.6. Выражения. Преобразований типов.	1	9–10	4		2	2	2,5	2			0,5	10							
3.7.	Тема 3.7. Операции языка: арифметические операции, операции отношения и логические операции, операция присваивания.	1	11–12	4	2		2	2,5	2			0,5	15							
3.8.	Тема 3.8. Функции. Определение и вызов функций. Глобальные данные и параметры. Параметры по умолчанию. Рекурсия. Перегрузка и шаблоны функций.	1	11–12	4		2	2	2,5	2			0,5	15							
3.9.	Тема 3.9. Массивы. Объявление и размещение элементов массива в оперативной памяти. Многомерные массивы. Обработка символьной информации. Массивы строк. Стандартные функции для работы со строками.	1	13–14	5	2		3	2,5	2			0,5	15							

№ п/п	Наименование разделов и тем дисциплины (модуля)	Семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)								Формы текущего контроля успеваемости (по неделям семестра)								
				Аудиторная работа				Самостоятельная работа				Собеседование	Коллоквиум	Проверка тестов	Проверка контрольн. работ	Проверка реферата	Проверка эссе и иных твор- ческих работ	курсовая работа (проект)	др.	
				Всего	Лекция	Практические занятия	Лабораторные занятия	Всего	Подготовка к аудиторным занятиям	Реферат, эссе и др.	Курсовая работа (проект)									Подготовка к зачету
3.10.	Тема 3.10. Указатели и ссылки. Связь между именами, адресами и значениями переменных. Арифметические операции и указатели. Ссылки.	1	13–14	5		2	3	2,5	2			0,5	15							
3.11.	Тема 3.11. Указатели и массивы.	1	15–16	4	2		3	2	2				15							
3.12.	Тема 3.12. Указатели и ссылки как параметры. Передача массивов в качестве параметров.	1	15–16	4		2	3	2,5	2			0,5	15							
	Обзорная лекция по C/C++	1	17	2	2	2														
	<i>Курсовая работа (проект)</i>																			
	<i>Подготовка к экзамену</i>																			
	Общая трудоемкость, в часах			72	18	18	36	36	32			4	Промежуточная аттестация							
												Форма		Семестр						
												Зачет		1						
												Экзамен		–						

Общая трудоемкость дисциплины за второй семестр составляет **5** зачетных единиц, **180** часа.

№ п/п	Наименование разделов и тем дисциплины (модуля)	Семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)									Формы текущего контроля успеваемости (по неделям семестра)							
				Аудиторная работа				Самостоятельная работа					Собеседование	Коллоквиум	Проверка тестов	Проверка контрол.н. работ	Проверка реферата	Проверка эссе и иных творческих работ	курсовая работа (проект)	др.
				Всего	Лекция	Практические занятия	Лабораторные занятия	Всего	Подготовка к аудиторным занятиям	Реферат, эссе и др.	Курсовая работа (проект)	Подготовка к экзамену								
4.	Раздел 4. Основы программирования на языке C/C++	2	1–12	46	12	10	24	52	52			26								
4.1.	Тема 4.1. Динамические переменные. Операторы и функции управления динамической памятью.	2	1–2	4	2		2	4	4			2	5							
4.2.	Тема 4.2. Выделение памяти под одномерный динамический массив и ее удаление. Обращение к элементам одномерного динамического массива	2	1–2	4		2	2	4	4			2	5							
4.3.	Тема 4.3. Выделение памяти под многомерный динамический массив и ее удаление. Обращение к элементам многомерного динамического массива.	2	3–4	4	2		2	4	4			2	5							
4.4.	Тема 4.4. Типичные ошибки при работе с динамической памятью.	2	3–4	4		2	2	4	4			2	5							
4.5.	Тема 4.5. Ввод/вывод в файл в C. Основные функции для работы с файлами	2	5–6	4	2		2	4	4			2	10							

№ п/п	Наименование разделов и тем дисциплины (модуля)	Семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)								Формы текущего контроля успеваемости (по неделям семестра)								
				Аудиторная работа				Самостоятельная работа				Собеседование	Коллоквиум	Проверка тестов	Проверка контролн. работ	Проверка реферата	Проверка эссе и иных твор- ческих работ	курсовая работа (проект)	др.	
				Всего	Лекция	Практические занятия	Лабораторные занятия	Всего	Подготовка к аудиторным занятиям	Реферат, эссе и др.	Курсовая работа (проект)									Подготовка к экзамену
4.6.	Тема 4.6. Ввод/вывод в С++: потоки языка с++, пространство имен. Файловый ввод/вывод в С++: Работа с классом fstream, ofstream, ifstream.	2	5–6	4		2	2	4	4			2	10							
4.7.	Тема 4.7. Примеры алгоритмов обработки дискретной информации. Сортировки и рекурсия.	2	7–8	4	2		2	4	4			2	10							
4.8.	Тема 4.8. Структуры данных: структуры (structure), объединения (union), перечислимый тип (enumeration), поля битов (bit fields), typedef.	2	7–8	4		2	2	4	4			2	10							
4.9.	Тема 4.9. Указатели на структуры. Структуры со ссылками на себя. Структуры и функции. Формирование динамических конструкций. Списки, операции над списками, их реализация в Си. Очередь. Ее реализация в Си.	2	9–10	4	2		2	6	6			3	10							

№ п/п	Наименование разделов и тем дисциплины (модуля)	Семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)								Формы текущего контроля успеваемости (по неделям семестра)							
				Аудиторная работа				Самостоятельная работа				Собеседование	Коллоквиум	Проверка тестов	Проверка контролн. работ	Проверка реферата	Проверка эссе и иных твор- ческих работ	курсовая работа (проект)	др.
				Всего	Лекция	Практические занятия	Лабораторные занятия	Всего	Подготовка к аудиторным занятиям	Реферат, эссе и др.	Курсовая работа (проект)								
4.10.	Тема 4.10. Абстрактные типы данных. Стек. Команды работы со стеком. Использование стека для организации вложенных циклов. Директивы препроцессора.	2	9–10	4		2	2	6	6			3	10						
4.11	Тема 4.11. Классы и объекты. Инкапсуляция, наследование, полиморфизм. Конструкторы и деструкторы. Перегрузка операций для классов. Примеры программ.	2	11–12	6	2		4	8	8			4	15						
.5.	Раздел 5. Создание Windows приложения в Microsoft Visual Studio	2	11–17	26	9	9	8	20	20			10	15						
5.1.	Тема 5.1. Интегрированная среда программирования Microsoft Visual Studio. Создание простого приложения	2	11–12	6	2	2	2	5	5			3	15						

№ п/п	Наименование разделов и тем дисциплины (модуля)	Семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)									Формы текущего контроля успеваемости (по неделям семестра)									
				Аудиторная работа				Самостоятельная работа					Собеседование	Коллоквиум	Проверка тестов	Проверка контролн. работ	Проверка реферата	Проверка эссе и иных твор- ческих работ	курсовая работа (проект)	др.		
				Всего	Лекция	Практические занятия	Лабораторные занятия	Всего	Подготовка к аудиторным занятиям	Реферат, эссе и др.	Курсовая работа (проект)	Подготовка к экзамену										
5.2.	Тема 5.2. Создание многоформенного приложения. Работа с меню	2	13–14	6	2	2	2	5	5			2	15									
5.3.	Тема 5.3. Обработка массивов	2	13–14	6	2	2	2	5	5			3	15									
5.4.	Тема 5.4. Работа с графикой VS	2	15–16	6	2	2	2	5	5			2	15									
	Обзорная лекция по Microsoft Visual Studio	2	17	2	1	1																
	<i>Курсовая работа (проект)</i>																					
	<i>Подготовка к экзамену</i>																					
	Общая трудоемкость, в часах			72	18	18	36	108	72			36	Промежуточная аттестация									
												Форма			Семестр							
												Зачет			–							
												Экзамен			2							

4.2. Содержание дисциплины

№ п/п	Наименование раздела дисциплины	Содержание раздела
1	Основы архитектуры семейства 80x86.	1. Архитектура ЭВМ по фон Нейману. Общие сведения о семействе процессоров 80x86. Центральный процессор. Общая шина. Память: ОЗУ, ПЗУ. Кэш-память. Адресное пространство.
2	Этапы разработки программы. Структура алгоритма. Языки проектирования алгоритмов	1. Этапы решения задач на ЭВМ. Алгоритм. Свойства алгоритма. Способы (формы) представления алгоритма. 2. Алгоритмизация линейных, разветвляющихся и циклических процессов: объекты, операции и управляющие структуры алгоритмов, общая структура алгоритма. 3. Языки проектирования алгоритмов (низкого и высокого уровня). Тестирование. Транслятор, компилятор, интерпретатор.
3	Основы программирования на языке Си	1. Процесс построения исполняемой программы. Структура программа (директивы препроцессора, библиотеки функций, функция main). 2. Представление чисел в компьютере. Форматы данных для представления целых чисел и плавающих чисел: одинарный, двойной, расширенный. Сопроцессор. 3. Переменные и константы. Область действия и область видимости переменной. Базовые типы данных (допустимые диапазоны значений переменных различных типов). 4. Управляющие структуры языка C/C++: следование; выбор; повторение. 5. Ввод/вывод в C/C++. printf(), scanf(), cin, cout. Пространство имен. 6. Выражения. Преобразований типов. 7. Операции языка: арифметические операции, операции отношения и логические операции, операция присваивания. 8. Функции. Определение и вызов функций. Глобальные данные и параметры. Параметры по умолчанию. Рекурсия. Перегрузка и шаблоны функций. 9. Массивы. Объявление и размещение элементов массива в оперативной памяти. Многомерные массивы. Обработка символьной информации. Массивы строк. Стандартные функции для работы со строками. 10. Указатели и ссылки. Связь между именами, адресами и значениями переменных. Арифметические операции и указатели. Ссылки. 11. Указатели и массивы. 12. Указатели и ссылки как параметры. Передача массивов в качестве параметров.
4	Основы программирования на языке C/C++	1. Динамические переменные. Операторы и функции управления динамической памятью. 2. Выделение памяти под одномерный динамический массив и ее удаление. Обращение к элементам одномер-

		<p>ного динамического массива</p> <p>3. Выделение памяти под многомерный динамический массив и ее удаление. Обращение к элементам многомерного динамического массива.</p> <p>4. Типичные ошибки при работе с динамической памятью.</p> <p>5. Ввод/вывод в файл в С. Основные функции для работы с файлами</p> <p>6. Ввод/вывод в С++: потоки языка с++, пространство имен. Файловый ввод/вывод в С++: Работа с классом <code>fstream</code>, <code>ofstream</code>, <code>ifstream</code>.</p> <p>7. Примеры алгоритмов обработки дискретной информации. Сортировки и рекурсия.</p> <p>8. Структуры данных: структуры (<code>structure</code>), объединения (<code>union</code>), перечислимый тип (<code>enumeration</code>), поля битов (<code>bit fields</code>), <code>typedef</code>.</p> <p>9. Указатели на структуры. Структуры со ссылками на себя. Структуры и функции. Формирование динамических конструкций. Списки, операции над списками, их реализация в Си. Очередь. Ее реализация в Си.</p> <p>10. Абстрактные типы данных. Стек. Команды работы со стеком. Использование стека для организации вложенных циклов. Директивы препроцессора.</p> <p>11. Классы и объекты. Инкапсуляция, наследование, полиморфизм. Конструкторы и деструкторы. Перегрузка операций для классов. Примеры программ.</p>
5	Создание Windows приложения в Microsoft Visual Studio	<p>1. Интегрированная среда программирования Microsoft Visual Studio. Создание простого приложения</p> <p>2. Создание многоформенного приложения. Работа с меню.</p> <p>3. Обработка массивов: одномерные и двумерные массивы.</p> <p>4. Работа с графикой VS.</p>

5. Образовательные технологии

В процессе изучения дисциплины «Программирование для ЭВМ» предполагается использовать структурно-логические и интеграционные образовательные технологии, реализуемые посредством:

– лекций в виде вводных, текущих, обзорных и заключительно-обобщающих занятий;

– практических занятий с использованием методов «многократного повторения» (темы 3.8, 3.10, 4.2, 4.6, 4.10); по логике мышления – индуктивные, дедуктивные и репродуктивные.

– организации самостоятельной работы на основе личностно-

дифференцированного подхода планирования задания в виде воспроизводящей и частично-поисковой работ.

– организации текущего контроля знаний студентов методами: выполнения домашних заданий, оценки активности на практических занятиях и рейтинговой системы общей оценки знаний студентов.

Занятия, проводимые в интерактивных формах, с использованием интерактивных технологий составляют 30 % занятий.

В целях реализации индивидуального подхода к обучению студентов, осуществляющих учебный процесс по собственной траектории в рамках индивидуального рабочего плана, изучение данной дисциплины базируется на следующих возможностях: обеспечение внеаудиторной работы со студентами в том числе в электронной образовательной среде с использованием соответствующего программного оборудования, дистанционных форм обучения, возможностей интернет-ресурсов, индивидуальных консультаций и т.д.

6. Учебно-методическое обеспечение самостоятельной работы студентов.

Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

6.1. План самостоятельной работы студентов

№ нед.	Тема	Вид самостоятельной работы	Задание	Рекомендуемая литература	Количество часов
1 семестр					
1–10	Основы программирования на языке Си	Подготовка к практическим занятиям	Типовое задание № 1	П.7 а) 1–2, б) 1	2
11–15	Основы программирования на языке Си	Подготовка к практическим занятиям	Типовое задание № 2	П.7 а) 1–2, б) 1	2
2 семестр					
1–10	Основы программирования на языке Си	Подготовка к практическим занятиям	Типовое задание № 3	П.7 а) 1–2, б) 1	8

11-15	Создание Windows приложения в Microsoft Visual Studio	Подготовка к практическим занятиям	Типовое задание № 3	П.7 а) 1–2, б) 1	4
-------	---	------------------------------------	---------------------	------------------	---

Образец типового практического задания № 1

- Сколько в языке С++ существует базовых типов для представления действительного значения?
- Для чего используются описатели signed и unsigned ?
- Какие из ниже перечисленных описаний являются описаниями базовых типов данных
 - const double A
 - int *B
 - unsigned long n
 - char v[10]
- Верно ли решена задача: «значение целочисленной переменной с увеличить на 1; целочисленной переменной а присвоить значение, равное удвоенному значению переменной с.


```
int a, c; c = 5;
```

 - c ++ ; a = 2 * c;
 - a = 2 * c ++ ;
 - c += 1; a = c + c;
 - a = c ++ + c;
 - ++c; a = c + c;
 - a = ++ c + c;
 - a = c += 1 + c;
 - a = (c+=1)+c;
- Пусть


```
char c; short s; int i; unsigned u; signed char sc;
float f; double d; long lng; unsigned short us; long double ld;
```

 Определить тип выражений:


```
c - s / i u * 3 - 3.0 * u - i (5*lng - 'a')*(s + u / 2 )
```
- Верно ли решена задача: «найти сумму первых 100 натуральных чисел»?
 - i = 1; sum = 0;


```
for ( ; i <= 100; i++) sum += i;
```
 - sum = 0;


```
for ( i = 1; i <= 100;) sum += i++;
```
 - for (i = 1, sum = 0; i <= 100; sum += i++);

Образец типового практического задания № 2

- Эквивалентны ли следующие объявления функций:
 - double f (); и double f (void);
 - h (double x); и int h (double x);
- Какие из приведенных ниже примеров являются описаниями функции?
 - int rus (unsigned char r)


```
{ if (r>='A' && c<=' ') return 1; else return 0; }
```
 - long sum (int k, ...);
 - long sq (int k) {return k*k};
 - printf("hello");
 - int printf(char *, ...);
- Какие из приведенных ниже примеров являются вызовами функции?
 - int rus (unsigned char r)


```
{ if (r>='A' && c<=' ') return 1; else return 0; }
```
 - long sum (int k, ...);

c) `int printf(char *, ...);`

4. Допустимо ли в Си? Если "да" - опишите семантику этих действий; если "нет" - объясните почему.

```
#include <stdio.h>
main()
{ int i; int sum = 0;
  for ( i = 1; i <= 3; i++)
    { sum += i;
      { int i;
        for ( i = 1; i <= 5; i++)
          sum *= i;
        }
      }
  printf("sum=%d\n", sum);
}
```

Образец типового практического задания № 3

1. Укажите зарезервированное ключевое слово для высвобождения выделенной памяти!
2. Укажите зарезервированное ключевое слово для динамического выделения памяти!
3. С какой целью выполняется явное преобразование типов значений функций `malloc` (`calloc`) при объявлении массивов?
4. С какой целью используются квадратные скобки в операции освобождения динамической памяти, выделенной под массив: `delete [] mass;`? Что произойдет, если их не указать?
5. Как правильно открыть поток для чтения из файла `input.txt`? Допишите нужную строку

```
FILE *f;
```

//открыли поток для чтения. Допишите строку.....

```
fscanf(f,"%d %d",&n,&k);
```

```
fclose(f);
```

6. Как организовать чтение 2-х целых чисел из потока `read` в переменные `x` и `y`?

```
read.open("file.txt");
```

```
//ЧТЕНИЕ
```

```
//ЧТЕНИЕ
```

```
//ЧТЕНИЕ
```

```
read.close("file.txt");
```

6.2. Методические указания по организации самостоятельной работы студентов

Подготовка к аудиторным занятиям проводится посредством изучения курса лекций, основной и дополнительной литературы, а также решения предложенных задач.

Подготовка рефератов и докладов осуществляется с использованием дополнительной литературы.

Выполнение лабораторных работ осуществляется с использованием основной литературы и изучением курса лекций и практических занятий.

Подготовка к экзамену – изучение курса лекций, упражнения в решении типовых задач, изучение дополнительной литературы.

6.3. Материалы для проведения текущего и промежуточного контроля знаний студентов

Контроль освоения компетенций

№ п\п	Вид контроля	Контролируемые темы (разделы)	Компетенции, компоненты которых контролируются
Семестр 1			
1	Лабораторная работа № 1	Раздел 2, 3	ОПК-2, ПК-1, ПК-3
2	Лабораторная работа № 3	Раздел 2, 3	ОПК-2, ПК-1, ПК-3
3	Лабораторная работа № 3	Раздел 2, 3	ОПК-2, ПК-1, ПК-3
4	Лабораторная работа № 4	Раздел 2, 3	ОПК-2, ПК-1, ПК-3
5	Лабораторная работа № 5	Раздел 2, 3	ОПК-2, ПК-1, ПК-3
Семестр 2			
1	Лабораторная работа № 1	Раздел 2, 3, 4	ОПК-2, ПК-1, ПК-3
2	Лабораторная работа № 3	Раздел 2, 3, 4	ОПК-2, ПК-1, ПК-3
3	Лабораторная работа № 3	Раздел 2, 3, 4	ОПК-2, ПК-1, ПК-3
4	Лабораторная работа № 4	Раздел 2, 3, 4	ОПК-2, ПК-1, ПК-3
5	Лабораторная работа № 5	Раздел 2, 5	ОПК-2, ПК-1, ПК-3

Лабораторные работы (1 семестр)

Лабораторная работа № 1. «Изучение среды разработки Microsoft Visual Studio»

1. Создать консольное приложение в Microsoft Visual C++, которое выводит на экран сообщение «Hello World». Запустить программу на исполнение и проверить результат её работы.

2. Изменить программу следующим образом:

```

#include <stdio.h>
#include <conio.h>

void main()
{
    // Определение имени, типа и
    // первоначального значения переменных
    int a = 1;
    int b = 2;
    int c = 3;
    int d = 0;

    // Вывод на экран значений переменных
    printf("a = %d\n", a);
    printf("b = %d\n", b);
    printf("c = %d\n", c);

    // Переход на следующую строку экрана
    printf("\n");

    // Вычисление значения переменной d
    d = a + b;
    printf("a + b = %d\n", d);

    // Вычисление значения переменной d
    d = b - c;
    printf("b - c = %d\n", d);

    // Вычисление значения переменной d
    d = a + b - c;
    printf("a + b - c = %d\n", d);

    // Вывод сообщения перед выходом из программы
    printf("Press any key to continue");
    getch();
}

```

3. Запустить программу на исполнение и посмотреть результаты её работы.

4. Изучить средства отладки программ

4.1 Установить точки останова на вычислениях значений переменной *d*, запустить программу на выполнение (клавиша F5), проконтролировать остановку выполнения программы в указанном месте и продолжение выполнения программы (клавиша F5).

4.2 Запустить выполнение программы в пошаговом режиме (клавиша F10), выполнить по шагам все строки программы (клавиша F10).

4.3 Выполнить пункты 4.1 и 4.2 с наблюдением за значением переменной *d* в окне Отладчика.

Лабораторная работа № 2. «Линейные алгоритмы»

Разработать блок-схему алгоритма и программу на языке Си, реализующие решение одной задачи задания. Исходные данные задаются в программе константами. Для числа π использовать значение 3.1415. Использовать тип данных **float**.

Задание

1. Найти площадь произвольного пятиугольника со сторонами *a*, *b*, *c*, *d*, *e*.

- Дана сторона равностороннего треугольника. Найти площадь этого треугольника и радиусы вписанной и описанной окружностей.
- Найти периметр и площадь равнобедренной трапеции с основаниями a и b ($a > b$) и углом α при большем основании (угол дан в радианах).
- Даны координаты трех вершин треугольника (x_1, y_1) , (x_2, y_2) , (x_3, y_3) . Найти его периметр и площадь.
- Определить площадь фигуры, представляющей собой неправильный четырёхугольник со сторонами a, b, c, d , из которого вырезана вписанная в четырёхугольник окружность радиусом R .
- Даны две стороны a и b треугольника и угол между ними α . Найти остальные углы и сторону треугольника.

7. Вычислить $y = \sqrt{2\pi \left| \cos \frac{x}{2} \sin x \right|} + e^{-\frac{x^2}{2+a}} + 0,8 \operatorname{tg} \left(\pi + \frac{ax^2}{\pi + \sqrt{|x|}} \right)$

8. Вычислить $z = \frac{\ln \frac{|x^3 - y^3|}{x^2 + 6} + 2 \ln |x - y| + \sin^2 x}{\sqrt[3]{e^{-y} + ae^{-x}} - \sqrt{6,4a\pi + e^{-ax}}}$

9. Вычислить $z = \pi \operatorname{ctg} \frac{\sqrt{|x^5 + a|}}{e^{-x} + e^y} - 1,65e^{-ax} + 0,33 \ln \frac{x^2}{x^2 + \pi}$

10. Вычислить $y = \frac{|a \sin x^3| + \operatorname{tg}^3 x \cos \sqrt{|x - a|} + \sqrt[3]{\operatorname{arctg}^2} + 1,17\pi}{(x + a)^2 e^{-\sin x} + \ln \sqrt{|2x + 1|}}$

11. Вычислить $z = \operatorname{tg} \frac{\sqrt{|x^3 + 1|}}{2,2e^{-x} + 1,1e^x} - \pi^2 \ln \frac{\sqrt[3]{x^2 + \cos \frac{y + \pi}{x}}}{e^{-y} + 3,6e^{-x}}$

Лабораторная работа № 3. «Ветвления и циклы»

Разработать блок-схему алгоритма и программу на языке C/C++, реализующие решение задачи из задания.

Задание А Условные операторы

- Даны три целых числа. Возвести в квадрат отрицательные числа и в третью степень – положительные (число 0 не изменять).
- Из трех данных чисел выбрать наименьшее и наибольшее.

3. Перераспределить значения переменных X и Y так, чтобы в X оказалось меньшее из этих значений, а в Y – большее.
4. Даны две переменные целого типа: A и B . Если их значения не равны, то присвоить каждой переменной сумму этих значений, а если равны, то присвоить переменным нулевые значения.
5. Даны три переменные: X , Y , Z . Если их значения упорядочены по убыванию, то удвоить их; в противном случае заменить значение каждой переменной на противоположное.
6. Даны три переменные: X , Y , Z . Если их значения упорядочены по возрастанию или убыванию, то удвоить их; в противном случае заменить значение каждой переменной на противоположное.
7. Даны вещественные координаты точки, не лежащей на координатных осях OX и OY . Вывести номер координатной четверти, в которой находится данная точка.
8. На числовой оси расположены три точки: A , B , C . Определить, какая из двух последних точек (B или C) расположена ближе к A , и вывести эту точку и ее расстояние от точки A .
9. Даны четыре целых числа, одно из которых отлично от трех других, равных между собой. Вывести порядковый номер этого числа.
10. Дан номер некоторого года (положительное целое число). Вывести соответствующий ему номер столетия, учитывая, что, к примеру, началом 20 столетия был 1901 год.

Задание Б Оператор выбора

1. Дан номер месяца (1 – январь, 2 – февраль, ...). Вывести число дней в этом месяце для невисокосного года.
2. Арифметические действия над числами пронумерованы следующим образом: 1 – сложение, 2 – вычитание, 3 – умножение, 4 – деление. Дан номер действия и два числа A и B (B не равно нулю). Выполнить над числами указанное действие и вывести результат.
3. Единицы длины пронумерованы следующим образом: 1 – дециметр, 2 – километр, 3 – метр, 4 – миллиметр, 5 – сантиметр. Дан номер единицы длины и длина отрезка L в этих единицах (вещественное число). Вывести длину данного отрезка в метрах.
4. Робот может перемещаться в четырех направлениях ("С" – север, "З" – запад, "Ю" – юг, "В" – восток) и принимать три цифровые команды: 0 – продолжать движение, 1 – поворот налево, -1 – поворот направо. Дан символ S – исходное направление робота и число N – посланная ему команда. Вывести направление робота после выполнения полученной команды.
5. Элементы окружности пронумерованы следующим образом: 1 – радиус (R), 2 – диаметр (D), 3 – длина (L), 4 – площадь круга (S). Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данной окружности (в том же порядке). В качестве значения P_i использовать 3.14.

6. Элементы равнобедренного прямоугольного треугольника пронумерованы следующим образом: 1 – катет (a), 2 – гипотенуза (c), 3 – высота, опущенная на гипотенузу (h), 4 – площадь (S). Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данного треугольника (в обратном порядке).
7. Даны два целых числа: D (день) и M (месяц), определяющие правильную дату невисокосного года. Вывести значения D и M для даты, предшествующей указанной.
8. Дано целое число в диапазоне 20 – 69, определяющее возраст (в годах). Вывести строку — словесное описание указанного возраста, обеспечив правильное согласование числа со словом "год", например: 20 — "двадцать лет", 32 — "тридцать два года", 41 — "сорок один год".
9. Дано целое число в диапазоне 100 – 999. Вывести строку — словесное описание данного числа, например: 256 — "двести пятьдесят шесть", 814 — "восемьсот четырнадцать".
10. В восточном календаре принят 60-летний цикл, состоящий из 12-летних подциклов, обозначаемых названиями цвета: зеленый, красный, желтый, белый и черный. В каждом подцикле годы носят названия животных: крысы, коровы, тигра, зайца, дракона, змеи, лошади, овцы, обезьяны, курицы, собаки и свиньи. По номеру года вывести его название, если 1984 год был началом цикла — годом зеленой крысы.

Задание В Циклы

1. Разработать программу, вычисляющую среднюю заработную плату отдела. Концом ввода сумм зарплаты является значение 0.
2. Определить порядковый номер и величину самой большой цифры, которые вводятся по одной с клавиатуры. Концом ввода является число (-100000).
3. Составить программу, проверяющую знание таблицы умножения на 3. В конце программы выдать количество верных и неверных ответов.
4. Составить программу, которая определяет количество пятёрок, четвёрок, троек и двоек, полученных в группе за контрольную работу. Оценки вводятся по одной с клавиатуры, концом ввода является ввод оценки 0.
5. Найти сумму отрицательных значений функции и количество отрицательных значений функции $y = \frac{2 \cos x \sin \frac{x}{2}}{\sqrt{|x|} + 5}$ на интервале изменения значений аргумента от x_n до x_k с шагом h .
6. Определить, на сколько произведение членов натурального ряда кратных пяти в диапазоне значений от 1 до 57 больше среднего арифметического этих чисел.
7. Определить сумму квадратов и сумму квадратных корней членов натурального ряда в диапазоне значений от 5 до 55.

8. Определить средний возраст участников соревнований при условии, что возраст каждого участника вводится с клавиатуры по одному, концом ввода данных является ввод числа 0.
9. Дано вещественное число X и целое число N ($N > 0$). Вычислите выражение $1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^N}{N!}$ ($N! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot N$). Полученное число является приближенным значением функции \exp в точке X .
10. Дано целое число N (> 2) и две вещественные точки на числовой оси: A, B ($A < B$). Функция $F(X)$ задана формулой $F(X) = 1 - \sin(X)$. Вывести значения функции F в N равноотстоящих точках, образующих разбиение отрезка $[A, B]$: $F(A), F(A + H), F(A + 2H), \dots, F(B)$.

Лабораторная работа № 4. «Функции»

Напишите функцию, которая осуществляет обработку информации по одному из условий, заданных ниже. Главная программа (**main**) осуществляет ввод исходных данных, передает их в функцию и выводит результаты, полученные из функции.

- 1) Дано натуральное число N . Определить:
 - а) количество цифр 7 в этом числе;
 - б) произведение цифр заданного натурального числа;
 - в) определить число, получаемое выписыванием в обратном порядке цифр заданного натурального числа;
 - г) является ли заданное натуральное число палиндромом, т.е. таким, десятичная запись которого читается одинаково слева направо и справа налево;
 - д) сколько чисел натурального ряда необходимо перемножить, чтобы их произведение превысило заданное натуральное число;
 - е) является ли число простым. Идея алгоритма заключается в том, что необходимо делить число N на все целые числа k меньше его (k можно ограничить диапазоном изменения от 2 до \sqrt{N}). Если N не делится нацело на возможные значения k , то N – простое число.
- 2) Найти наибольший общий делитель для двух натуральных чисел m и n ($m \geq 0, n \geq 0, m > n$).
- 3) Функция вычисляет квадратный корень вещественного аргумента x с заданной точностью ϵ по итерационной формуле:

$$y_n = \frac{1}{2} \left(y_{n-1} + \frac{x}{y_{n-1}} \right),$$

где y_{n-1}, y_n – предыдущее и последующее приближения корня соответственно. Процесс вычислений прекращается, когда приближения станут отличаться друг от друга по абсолютной величине менее чем на величину ϵ .

- 4) Функция вычисляет $\sin(x)$ вещественного аргумента x по формуле:

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots + (-1)^n \cdot \frac{x^{2n+1}}{(2n+1)!} \text{ с точностью } 0,001$$

5) Функция вычисляет e^x вещественного аргумента x по формуле:

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!} \text{ с точностью } 0,001$$

Напишите функцию, которая осуществляет рекурсивную обработку информации по одному из условий, заданных ниже

$$6) y = \sum_i^{10} (2+i)$$

$$7) y = \sqrt{3 + \sqrt{6 + \sqrt{9 + \dots \sqrt{96 + \sqrt{99}}}}}$$

$$8) y = \frac{1}{1 + \frac{1}{3 + \frac{1}{5 + \frac{1}{7 + \frac{1}{9 + \frac{1}{11}}}}}}$$

9) Определить наибольший общий делитель по алгоритму Евклида (схема алгоритма из примера 2.7.1.2) для двух чисел A и B ($A \geq 0, B \geq 0, A > B$).

10) Найти n ($n > 2$) первых чисел Фибоначчи: 0, 1, 1, 2, 3, 5, ... Обозначим число как результат функции f (номер). Тогда последовательность чисел Фибоначчи можно рекурсивно определить так: $f(0) = 0, f(1) = 1, f(n) = f(n-1) + f(n-2)$.

Лабораторная работа № 5. «Массивы»

Разработать блок-схему алгоритма и программу на языке C/C++, реализующие решение задачи из задания.

1. Найти среднее арифметическое положительных элементов матрицы $Z(n,m)$, сформированной случайным образом, и количество её отрицательных элементов. Значения параметров n и m задаются с клавиатуры и подчиняются следующим условиям: $n \leq 12, m \leq 10$.
2. В матрице $Z(n,n)$, сформированной случайным образом, найти среднее арифметическое отрицательных элементов матрицы, расположенных в чётных строках. Значение параметра n задаётся с клавиатуры и подчиняется условию: $n \leq 12$.
3. В матрице $A(9,8)$, сформированной случайным образом, найти максимальный элемент каждой строки и поместить эти значения в вектор Y .
4. В матрице $C(7,6)$, сформированной случайным образом, расположить элементы каждой строки в порядке возрастания их величины.
5. В матрице $B(6,6)$, сформированной случайным образом, найти максимальный элемент каждого столбца и поместить эти значения в вектор X .

6. Произвести умножение матрицы $T(3,6)$ на матрицу $D(6,2)$. В полученном произведении найти положение минимального элемента. Матрицы сформированы случайным образом.
7. В матрице $C(7,6)$, сформированной случайным образом, расположить элементы каждого столбца в порядке убывания величины элементов.
8. Во введенном с клавиатуры тексте найти количество слов, начинающихся с гласной буквы.
9. Дана матрица размера 7×9 . Поменять местами ее строки так, чтобы их минимальные элементы образовывали возрастающую последовательность.

Лабораторные работы (2 семестр)

Лабораторная работа № 1. «Динамическая память»

1. Создать квадратную матрицу, размер которой вводится с клавиатуры. Заполнить матрицу таким образом, чтобы числа возрастали по мере приближения к центру матрицы. Вывести результат на экран. Использовать функцию `malloc`.

2. Создать два вектора, размерность которых вводится с клавиатуры. Заполнить вектора случайными числами. Найти сумму и разность этих векторов, а также их векторное произведение. Результат записать на экран. Использовать функцию `calloc`.

3. Создать две квадратные матрицы, размерность которых вводится с клавиатуры. Заполнить матрицы случайными числами. Выполнить сложение матриц и транспонировать результат. Результат вывести на экран. Использовать оператор `new`.

4. Создать прямоугольную матрицу, размерность которой вводится с клавиатуры. Заполнить матрицу случайными четными числами. Вывести на экран квадратную матрицу, получаемую отсечением первых лишних строк (или столбцов). Результат вывести на экран. Использовать оператор `new`.

5. Создать два вектора, размерность которых вводится с клавиатуры. Заполнить вектора случайными числами. Проверить векторы на коллинеарность. Вычислить скалярное произведение векторов. Результат вывести на экран. Использовать функцию `calloc`.

6. Создать матрицу, размерность которой вводится с клавиатуры. Заполнить матрицу случайными числами. Увеличить размерность матрицы на величину, которая вводится с клавиатуры. Заполнить новые элементы матрицы единицами. Результат вывести на экран. Использовать оператор `new`.

7. Создать квадратную матрицу, размерность которой вводится с клавиатуры. Заполнить элементы матрицы расположенные выше главной диагонали числами натурального ряда (1, 2, 3, ...), остальные нулями. Результат вывести на экран. Использовать оператор `new`.

8. Создать прямоугольную матрицу, размерность которой вводится с клавиатуры. Требуется дополнить ее до квадратной. Заполнить новые элементы матрицы нулями. Результат вывести на экран. Использовать функцию `malloc`.

9. Создать три вектора, размерность которых вводится с клавиатуры. Заполнить вектора случайными нечетными числами. Проверить векторы на компланарность и вычислить объем построенной на них пирамиды. Результат вывести на экран. Использовать функцию `calloc`.

10. Создать матрицы, размерности которых вводятся с клавиатуры (число столбцов должно быть одинаковое). Заполнить матрицы случайными числами. Из первой матрицы требуется удалить четные строки и добавить эти строки в конец второй матрицы. Результат вывести на экран. Использовать функцию `malloc`.

Лабораторная работа № 2. «Библиотеки ввода/вывода и работа с файлами»

Выполнить лабораторную работу № 1, и модифицировать согласно указанным ниже положениям. Предусмотреть различные варианты ввода данных, а именно, при запуске программы появляется приглашение на ввод начальных данных: ввод данных (с клавиатуры); чтение данных (из файла); `rand()`. Вывод данных и промежуточных результатов осуществить в другой файл. Требуется засечь время выполнения всей программы (без пункта 2, при ручном вводе). Построить график зависимости времени выполнения от размерности входных матриц (векторов). В этом случае время определяется для варианта ввода данных – `rand()`. Построение графика можно осуществить в MS Excel (или любой другой среде: MathCad, Maple и т.д.).

Лабораторная работа № 3. «Структуры»

Требуется разработать программу, выполняющую следующие функции:

- ввод данных (с клавиатуры);
- чтение данных (из файла);
- вывод данных (на экран);
- сортировка данных по заданному атрибуту (вычисляемому полю);

Варианты заданий отличаются структурой данных, алгоритмом сортировки и другими параметрами.

№ Варианта	Предметная область	Входные данные	Вычисляемое поле
1	Меню	Наименование блюда, вес, цена, время приготовления	Цена со скидкой (10 %)
2	Расчет оплаты за детский сад	ФИО ребенка, возраст ребенка, стоимость 1-го дня (руб.), количество посещений (дней)	Итого (руб.)
3	Расписание игр	Название команды, дата игры, цена билета (для взрослых), место проведения	Цена билета (детского, скидка 50 %)
4	Прокат автомобиля	ФИО клиента, марка авто, дата взятия, дата возврата, цена за час (руб.)	Итого за прокат (руб.)
5	Склад	Наименование товара, инвентарный номер, дата поставки, закупочная цена (руб.)	Отпускная цена (руб.) с учетом 20 % надбавки
6	Гостиница	ФИО клиента, № комнаты, количество суток проживания, цена за сутки (руб.)	Итого за все время проживания (руб.)

7	Аптека	Название препарата, изготовитель, срок годности, цена (руб.)	Цена со скидкой (8 %)
8	Численность населения	Название страны, численность населения, площадь, язык	Плотность населения (равна отношению численности к площади)
9	Расчет коммунальных услуг	ФИО квартиросъемщика, адрес, количество прописанных человек, тариф на человека (руб.)	Итого (руб.)
10	Книга	ФИО автора, место издание, год издания, цена (руб.)	Цена (студентам скидка 30 %)

Алгоритмы сортировок: четный вариант – пузырьковая сортировка; нечетный вариант – метод прямого выбора. Схема работы программы:

1 шаг. При запуске программы появляется приглашение на ввод данных:

- 1 – ввод данных (с клавиатуры);
- 2 – чтение данных (из файла);

При нажатии на соответствующую цифру запускается соответствующая функция (именно реализованная как функция, например

```
void имя_функции (struct имя_структуры_в_которую_будет_происходить_ввод)
{
};
```

2 шаг. По всем строкам структуры заполняется вычисляемое поле.

3 шаг. Далее введенная структура выводится на экран (это опять реализуется в виде функции). Кроме того на экране должно присутствовать вычисляемое поле (для каждой записи).

4 шаг. Запускается функция сортировки (по возрастанию) всех записей структуры по вычисляемому полю.

5 шаг. Далее введенная структура опять выводится на экран (это запускается та же функция что и на втором шаге). Кроме того на экране должно присутствовать вычисляемое поле (для каждой записи). *Теперь строки должны были поменяться местами.*

Пример создаваемой структуры

10	Книга	ФИО автора, место издание, год издания, цена (руб.)	Цена (студентам скидка 30 %)
----	-------	---	------------------------------

```
struct book
{
char FIO[30]; //ФИО автора
char State[10]; //место издание
```

```

int Year;//год издания
float Price;//цена
float PriceSale;//цена со скидкой. Это поле не заполняется
//первоначально. Оно вычисляется позже.
};
struct book TABLE[5]; //создается массив TABLE[5] типа struct book

```

Лабораторная работа № 4. «Классы»

Этапы задания:

1 этап. Модульное программирование. Разработать библиотеку функций (h-файл) для выполнения:

- чтения базы данных из файла; //read()
- печати на экран; //print()
- добавления и удаления строк базы данных; //add(), del()
- записи базы данных в файл; //record()
- сортировка по вычисляемому полю.//sort()

2 этап. Создать класс на основе структуры в лабораторной работе № 3.

Пример создаваемой структуры

10	Книга	ФИО автора, место издание, год издания, цена (руб.)	Цена (студентам скидка 30 %)
----	-------	--	---------------------------------

```

struct book
{
char FIO[30];//ФИО автора
char State[10];//место издание
int Year;//год издания
float Price;//цена
float PriceSale;//цена со скидкой. Это поле не заполняется
//первоначально. Оно вычисляется позже.
};

```

Создание класса

```

class book {
privat://закрытые
//Данные
char FIO[30];//ФИО автора
char State[10];//место издание
int Year;//год издания
float Price;//цена
float PriceSale;//цена со скидкой. Это поле не заполняется
//первоначально. Оно вычисляется позже.
//Методы
public://открытые
book();//Конструктор по умолчанию.

```

```

        //Конструктор НЕ возвращает значения!
тип read(аргументы); //чтения базы данных из файла
тип print(аргументы); //печать на экран
тип add(аргументы); // добавление строк базы данных
тип del(аргументы); // удаление строк базы данных
тип record (аргументы); // запись базы данных в файл
тип sort (аргументы); // сортировка по вычисляемому полю
    }

```

Лабораторная работа № 5. «Windows приложения»

Индивидуальное задание к разделу 1

Вариант 1. Организуйте вычисление следующих величин: 1. Сумму двух целых чисел; 2. Разность двух целых чисел.

Вариант 2. Организуйте вычисление следующих величин: 1. Произведение двух целых чисел; 2. Частное двух целых чисел.

Вариант 3. Организуйте вычисление следующих величин: 1. Среднее арифметическое двух целых чисел; 2. Средне геометрическое двух целых чисел.

Вариант 4. Организуйте вычисление следующих величин: 1. Полусумму двух целых чисел; 2. Полуразность двух целых чисел.

Вариант 5. Организуйте вычисление следующих величин: 1. Объем куба, заданного своим ребром; 2. Площадь боковой поверхности куба, заданного своим ребром.

Вариант 6. Организуйте вычисление следующих величин: 1. Даны радиус основания и высота цилиндра, найти его объем; 2. Даны радиус основания и высота цилиндра, найти его площадь поверхности.

Вариант 7. Организуйте вычисление следующих величин: 1. Даны основания a , b трапеции и высота h , найти площадь трапеции; 2. Даны основания a , b трапеции и высота h , найти периметр трапеции.

Вариант 8. Организуйте вычисление следующих величин: 1. Даны три стороны треугольника. Найти его периметр; 2. Даны три стороны треугольника. Найти его площадь.

Вариант 9. Организуйте вычисление следующих величин: 1. Даны радиус основания и высота конуса, найти его объем; 2. Даны радиус основания и высота конуса, найти его площадь поверхности.

Вариант 10. Организуйте вычисление следующих величин: 1. Даны сторона a ромба и угол между ними, найти площадь ромба; 2. Даны сторона a ромба и угол между ними, найти периметр ромба.

Индивидуальное задание к разделу 2

Вариант 1. Главное меню:
 Файл: Сумма, произведение.
 Справка: Об авторе

Контекстное меню: Очистить, обнулить.

Вариант 2. Главное меню:

Файл: Разность, частное.

Справка: О программе

Контекстное меню: Очистить, установить в 1.

Вариант 3. Главное меню:

Файл: Среднее арифметическое, разность.

Справка: Об авторе

Контекстное меню: Очистить, обнулить.

Вариант 4. Главное меню:

Файл: Полуразность, произведение.

Справка: О программе

Контекстное меню: Очистить, удвоить результат.

Вариант 5. Главное меню:

Файл: Полу-сумма, полу-произведение.

Справка: Об авторе

Контекстное меню: Очистить, установить в 1.

Вариант 6. Главное меню:

Файл: Среднее геометрическое, произведение.

Справка: Основные формулы

Контекстное меню: Очистить, уменьшить результат на 1.

Вариант 7. Главное меню:

Файл: Сумма, уменьшить на 2 произведение.

Справка: Об авторе

Контекстное меню: Очистить, обнулить.

Вариант 8. Главное меню:

Файл: Произведение, частное.

Справка: О программе

Контекстное меню: Очистить, уменьшить на 5.

Вариант 9. Главное меню:

Файл: Разность, частное.

Справка: О программе

Контекстное меню: Очистить, уменьшить на 10.

Вариант 10. Главное меню:

Файл: Удвоенное произведение, сумма.

Справка: Основные формулы

Контекстное меню: Очистить, увеличить на 3.

Индивидуальное задание к разделу 3

Вариант 1

1. Вычислить минимальный элемент одномерного массива.
2. Вычислить максимальный элемент двумерного массива.

Вариант 2

1. В одномерном массиве вычислить сумму отрицательных элементов.

2. В двумерном массиве вычислить минимальный элемент.

Вариант 3

1. В одномерном массиве найти количество одинаковых чисел.
2. В двумерном массиве вычислить сумму четных элементов.

Вариант 4

1. Определить количество соседств двух чисел одного знака в одномерном массиве.
2. В двумерном массиве определить максимальный среди элементов, стоящих выше главной диагонали.

Вариант 5

1. В одномерном массиве вычислить сумму положительных элементов.
2. В двумерном массиве найти номер строки, содержащей максимальный элемент.

Вариант 6

1. Вычислить максимальный элемент одномерного массива.
2. В двумерном массиве вычислить сумму элементов, стоящих ниже главной диагонали.

Вариант 7

1. В одномерном массиве найти сумму минимального и максимального элементов.
2. В двумерном массиве найти номер столбца, содержащего минимальный элемент.

Вариант 8

1. В одномерном массиве найти сумму элементов, больших по модулю 10.
2. В двумерном массиве найти количество положительных элементов.

Вариант 9

1. В одномерном массиве найти сумму элементов, заключенных между минимальным и максимальным.
2. В двумерном массиве найти номер столбца с наибольшей суммой элементов.

Вариант 10

1. В одномерном массиве найти сумму четных элементов.
2. В двумерном массиве вывести номер столбца с наибольшей суммой элементов.

Индивидуальное задание разделу 4

Вариант 1

1. Нарисуйте цветной заполненный круг указанным радиусом. Действия над объектами: вывод изображения на экран и удаление изображения (в графическом режиме), перемещение объекта по экрану на введенную пользователем величину.
2. Построить график функции, изменяя параметр X и с шагом h. Ввод исходных данных производится пользователем.

$$y(x) = \frac{2 \cos(x)}{0.5 + \sin^2 x}$$

3. Построить диаграмму изменения курса доллара, данные вводятся пользователем в список.

Вариант 2

1. Нарисуйте цветной заполненный круг. Действия над объектами: изменение цвета границы и заполнения, удаление.
2. Построить график функции, изменяя параметр X и с шагом h . Ввод исходных данных производится пользователем.

$$y(x) = \frac{\sqrt{6+x}}{x^2+2}$$

3. Построить диаграмму изменения курса евро, данные вводятся пользователем в список.

Вариант 3

1. Нарисуйте цветной заполненный прямоугольник. Действия над объектами: вывод изображения на экран и удаление изображения (в графическом режиме), перемещение объекта по экрану на введенную пользователем величину.
2. Построить график функции, изменяя параметр X и с шагом h . Ввод исходных данных производится пользователем.

$$y(x) = \frac{1 + \sin(x)}{x - \frac{2}{x}}$$

3. Построить диаграмму изменения дохода фирмы, данные вводятся пользователем в список.

Вариант 4

1. Нарисуйте цветной заполненный прямоугольник. Действия над объектами: изменение цвета и вида заполнения, изменение размера на величину, заданную пользователем.
2. Построить график функции, изменяя параметр X и с шагом h . Ввод исходных данных производится пользователем.

$$y(x) = |\cos(x)| - \frac{x^2}{2}$$

3. Построить диаграмму изменения количества клиентов фирмы, данные вводятся пользователем в список.

Вариант 5

1. Нарисуйте цветной заполненный треугольник. Действия над объектами: вывод изображения на экран и удаление изображения (в графическом режиме), перемещение объекта по экрану на введенную пользователем величину.
2. Построить график функции, изменяя параметр X и с шагом h . Ввод исходных данных производится пользователем.

$$y(x) = \ln(2^x) - \sin^2(x)$$

3. Построить диаграмму изменения количества заказов фирмы, данные вводятся пользователем в список.

Вариант 6

1. Нарисуйте цветной заполненный треугольник. Действия над объектами: изменение цвета и вида заполнения.

2. Построить график функции, изменяя параметр X и с шагом h . Ввод исходных данных производится пользователем.

$$y(x) = \sqrt{10 * (\sqrt{x} + x)}$$

3. Построить диаграмму изменения количества автомобилей, приехавших на заправочную станцию, данные вводятся пользователем в список.

Вариант 7

1. Нарисуйте цветной отрезок прямой. Пользователь задает координаты конца отрезка и цвет.

2. Построить график функции, изменяя параметр X и с шагом h . Ввод исходных данных производится пользователем.

$$y(x) = \frac{e^x * \cos(x)}{x + 1}$$

3. Построить диаграмму изменения количества продукции, произведенной фирмой, данные вводятся пользователем в список.

Вариант 8

1. Нарисуйте цветную ломаную линию. Действия над объектами: вывод изображения на экран и удаление изображения (в графическом режиме), изменение цвета. Количество звеньев ломанной $N=5$.

2. Построить график функции, изменяя параметр X и с шагом h . Ввод исходных данных производится пользователем.

$$y(x) = x^2 - \sqrt{\frac{1}{x}}$$

3. Построить диаграмму изменения количества клиентов, обслуженных в магазине, данные вводятся пользователем в список.

Вариант 9

1. Нарисуйте цветной заполненный эллипс. Действия над объектами: вывод изображения на экран и удаление изображения (в графическом режиме), перемещение объекта по экрану на введенную пользователем величину.

2. Построить график функции, изменяя параметр X и с шагом h . Ввод исходных данных производится пользователем.

$$y(x) = \sqrt{10 * (\sqrt{x} + x)}$$

3. Построить диаграмму изменения количества деталей, произведенных цехом, данные вводятся пользователем в список.

Вариант 10

1. Нарисуйте цветной заполненный эллипс. Действия над объектами: изменение цвета и вида заполнения, изменение одного из радиусов на величину, заданную пользователем.

2. Построить график функции, изменяя параметр X и с шагом h . Ввод исходных данных производится пользователем.

$$y(x) = \frac{e^x * \cos(x)}{x + 1}$$

3. Построить диаграмму изменения количества продукции, произведенной фирмой, данные вводятся пользователем в список.

Темы рефератов

1. Директивы препроцессора.
2. Абстрактные типы данных: очередь
3. Абстрактные типы данных: стек
4. Абстрактные типы данных: списки

Вопросы к зачету по дисциплине «Программирование ЭВМ» (1 семестр)

Первый семестр

1. ЭВМ. Основные принципы формирования ЭВМ по фон Нейману
2. Этапы решения задач на ЭВМ.
3. Алгоритм. Свойства алгоритма. Способы (формы) представления алгоритма.
4. Алгоритмизация линейных, разветвляющихся и циклических процессов.
5. Типы алгоритмов. Языки программирования (низкого и высокого уровня).
6. Транслятор, компилятор, интерпретатор. Язык C/C++. Процесс построения исполняемой программы.
7. Библиотеки функций и заголовочные файлы языка C. Главная программа (main). Принцип работы программы (подключение библиотек, запуск функции main и т.д.).
8. Представление целых чисел в компьютере.
9. Представление вещественных чисел в компьютере.
10. Переменные. Типы (базовые) целочисленных данных: char, int. Модификаторы (signed, unsigned, long, short).
11. Переменные. Типы (базовые) вещественных данных (float и т.д.). Операция sizeof.
12. Управляющие структуры языка C/C++. Структуры выбора (условные операторы if, if-else и switch. Операция условие ?:).
13. Управляющие структуры языка C/C++. Структуры повторения (while и do-while. Операторы break и continue).
14. Управляющие структуры языка C/C++. Структуры повторения (for. Бесконечный цикл и условие его прерывания). Вложенные циклы.
15. Ввод и вывод (библиотека stdio.h). Функция printf(). Функция scanf().
16. Выражения. Приведение типов. Объявление констант (форма объявления). Спецификатор const.
17. Операции языка C (классификация по числу операндов). Арифметические операции. Операции отношения и логические операции. Операция присваивания.
18. Массивы. Объявление массива в программе. Доступ к элементам массива. Двумерные массивы. Инициализация массивов.
19. Массивы символов. Строки. Функции scanf() и gets(). Функции printf() или

puts()).

20. Функции для работы со строками: strcpy(), strcat(), strlen(), strcmp().

21. Функции в языке С. Описание функции. Оператор return. Объявление функции (прототип функции).

22. Область действия и область видимости (локальная, глобальная, формальная переменная).

23. Указатели. Имена указателей. Операторы & (взятия адреса) и * (разыменования). Обращение к данным через указатели.

24. Ссылка. Имена ссылок. Использование оператора адреса (&) при работе со ссылками. Обращение к данным через ссылки.

25. Параметры и аргументы (аргументы передаются по значению). Возвращение нескольких значений. Передача аргументов функций как указателей. Возвращение значений с помощью указателей.

26. Параметры и аргументы (аргументы передаются по значению). Возвращение нескольких значений. Передача аргументов функций как ссылок. Возвращение значений с помощью ссылок.

27. Операции над указателями. Связь указателей и массивов. Массивы как аргументы функций.

28. Объявление переменных (форма объявления). Инициализация переменных. Локальная и глобальная переменная; переменная как формальный параметр

Вопросы к экзамену по дисциплине «Программирование ЭВМ»

(2 семестр)

1. Одномерные динамические массивы: объявление, выделение памяти в С (malloc, calloc), освобождение памяти в С, обращение к элементам одномерного динамического массива.

2. Одномерные динамические массивы: объявление, выделение памяти в С++ (new), освобождение памяти в С++, обращение к элементам одномерного динамического массива.

3. Двумерные динамические массивы: объявление, выделение памяти в С (malloc, calloc), освобождение памяти в С, обращение к элементам двумерного динамического массива.

4. Двумерные динамические массивы: объявление, выделение памяти в С++ (new), освобождение памяти в С++, обращение к элементам двумерного динамического массива.

5. Преимущества и недостатки использования динамической памяти. Типичные ошибки при работе с динамической памятью.

6. Ввод и вывод в С. Стандартный ввод/вывод в С. Ввод/вывод в файл в С. Основные функции для работы с файлами.

7. Ввод/вывод в С++. Стандартный ввод/вывод в С++ (cin и cout). Файловый ввод/вывод в С++.

8. Пространства имен. Потоки. Работа с классом fstream. Работа с классом ofstream. Работа с классом ifstream.

9. Рекурсия в С++. Основные понятия и примеры.

10. Основные правила и принципы организации измерения времени в С/С++ (основные функции и их возможности).

11. Типы, определяемые пользователем. Структура.

12. Типы, определяемые пользователем. Поля битов. Доступ к отдельному биту.
13. Типы, определяемые пользователем. Объединения.
14. Типы, определяемые пользователем. Перечислимый тип. Переименование типов – typedef.
15. Препроцессор языка Си. Директивы условной компиляции.
16. Ограничения накладываемые на структуру в С, и дополнительные возможности структуры в С++.
17. Динамические структуры данных. Стек. Список. Дека.
18. Объектно-ориентированное программирование. Класс. Определение класса.
19. Инкапсуляция. Наследование. Полиморфизм. Конструкторы и деструкторы.
20. Создание Windows приложения в Microsoft Visual Studio. Функции-обработчики событий. Класс Form и компоненты: CheckBox, TextBox, Button, Label; ListBox, DataGridView; pictureBox.

Задания к зачету (1 семестр)

1. Дана квадратная матрица порядка M . Найти произведение положительных элементов, располагающихся ниже главной диагонали.
2. Дана целочисленная матрица размера $M \times N$. Различные строки матрицы назовем похожими, если совпадают множества чисел, встречающихся в этих строках. Найти количество строк, похожих на первую строку.
3. Дана квадратная матрица порядка M . Найти максимальный по модулю отрицательный элемент среди элементов, располагающихся выше главной диагонали.
4. Дана целочисленная матрица размера $M \times N$. Найти количество положительных и отрицательных элементов матрицы и количество нулей.
5. Дана квадратная матрица порядка M . Зеркально отразить ее элементы относительно (побочной диагонали) матрицы.
6. Задан двумерный массив размерности $N \times N$. Получить среднее арифметическое отрицательных элементов массива, сумма позиции которых нечетна.
7. Дана строка (из нескольких слов, разделенных пробелами). Вывести самое короткое слово в строке.
8. Дана строка (из нескольких слов, разделенных пробелами). Удалите в ней второе слово и выведите на экран результат.
9. В одномерном массиве, состоящем из p целых элементов, вычислить произведение элементов массива, расположенных между максимальным и первым элементами.
10. В одномерном массиве, состоящем из p целых элементов, вычислить сумму элементов массива, расположенных между первым и последним нулевыми элементами.
11. Дана целочисленная прямоугольная матрица. Определить количество строк, не содержащих ни одного нулевого элемента.

12. Дана целочисленная прямоугольная матрица. Определить количество столбцов, содержащих хотя бы один нулевой элемент
13. Дана целочисленная квадратная матрица. Определить произведение элементов в тех строках, которые не содержат отрицательных элементов
14. Для заданной матрицы размером 8 на 8 найти такие k , что k -я строка матрицы совпадает с k -м столбцом.
15. Дана целочисленная квадратная матрица. Определить минимум среди сумм элементов диагоналей, параллельных главной диагонали матрицы.
16. Дана целочисленная прямоугольная матрица. Определить номер столбца, в которой находится самая длинная серия одинаковых элементов.
17. Дана целочисленная прямоугольная матрица. Определить количество отрицательных элементов в тех строках, которые содержат хотя бы один нулевой элемент.
18. Задана целочисленная матрица A размерностью $N \times M$ и массив $X(N)$. Обнулить строки матрицы с номером K , для которого $X[K] \geq 0$.
19. Найти минимальный элемент, расположенный на главной диагонали матрицы A , а из элементов строки и столбца, на пересечении которых он находится, сформировать одномерный массив B
20. Найти «седловую» точку матрицы – такой элемент A_{ij} , который является наибольшим в строке i и наименьшим в столбце j . Если такого элемента нет, то выдать соответствующее сообщение

Задания к экзамену (2 семестр)

В каждом из представленных заданий память под массив (матрицу) должна выделяться динамически.

1. Требуется считать из файла действительную прямоугольную матрицу размерности $N \times M$. Найти сумму наибольших значений элементов ее строк. Результат вывести на экран.
2. Требуется считать из файла целочисленную прямоугольную матрицу размерности $N \times M$. Определить количество строк, не содержащих ни одного положительного элемента. Результат вывести на экран.
3. Для действительной матрицы размером N на N , заданной с клавиатуры необходимо найти такие k , что k -я строка матрицы совпадает с k -м столбцом. Матрицу и результат вывести в файл.
4. С клавиатуры задается целочисленная прямоугольная матрица размерности $N \times M$. Определить количество столбцов, содержащих хотя бы один нулевой элемент. Матрицу и результат вывести в файл.
5. С клавиатуры задается действительная квадратная матрица. Определить минимум среди сумм элементов диагоналей, параллельных главной диагонали матрицы. Результат вывести в файл.
6. Из файла требуется считать действительную квадратную матрицу размерности $N \times N$. Определить произведение элементов в тех строках, которые не содержат отрицательных элементов. Результат вывести на экран.

7. Требуется считать из файла действительную прямоугольную матрицу размерности $N \times M$. Определить номер столбца, в которой находится самая длинная серия одинаковых элементов. Результат вывести на экран.

8. Задана целочисленная матрица A размерностью $N \times M$ (считать из файла) и массив X размерности N (задан с клавиатуры). Требуется обнулить строки матрицы с номером K , для которого $X[K] \geq 0$. Результат вывести на экран.

9. Из файла требуется считать целочисленную квадратную матрицу A размерности N . Найти «седловую» точку матрицы – такой элемент $A_{i,j}$, который является наибольшим в строке i и наименьшим в столбце j . Если такого элемента нет, то выдать соответствующее сообщение (на экран).

10. С клавиатуры задается целочисленная квадратная матрица размерности N . Если матрица содержит больше столбцов, с положительной суммой элементов, чем строк с положительной суммой элементов, то вывести в файл слово «столбец», иначе «строка».

11. С клавиатуры требуется считать действительную квадратную матрицу A размерности N . Среди элементов, расположенных выше главной диагонали матрицы A , найти те элементы, которые удовлетворяют условию $-10 \leq A[i, j] \leq 10$, и сформировать из них одномерный массив B . Результат вывести в файл.

12. Из файла требуется считать действительную квадратную матрицу A размерности N . Найти минимальный элемент, расположенный на главной диагонали матрицы A , а из элементов строки и столбца, на пересечении которых он находится, сформировать одномерный массив B . Результат вывести на экран.

13. Задана целочисленная матрица A размерностью $N \times M$ (считать с клавиатуры). Требуется найти наименьший элемент матрицы A и записать нули в ту строку и столбец, где он находится. Результат вывести в файл.

7. Учебно-методическое и информационное

обеспечение дисциплины «Программирование для ЭВМ»

а) основная литература

1/ Основы программирования на языке C++ : учебное пособие / С.В. Шибанов, Е. Н. Дубровина ; под ред. Б. Г. Хмелевского ; Пенз. гос. ун-т. - Пенза : Изд-во Пенз. гос. ун-та, 2008. - 112 с. http://kleopatra.pnzgu.ru/cgi-bin/irbis64r_91/cgiirbis_64.exe?P21DBN=KATL&I21DBN=KATL_PRINT&S21FMT=fullw_print&C21COM=F&Z21MFN=9655 (119 экземпляров)

2. Основы языка C в примерах [Текст] : учебное пособие / Г. К. Чистова, Ю. С. Акимова, В. Б. Механов ; Пенз. гос. ун-т. - Пенза : Изд-во Пенз. гос. ун-та, 2010. - 80 с. http://kleopatra.pnzgu.ru/cgi-bin/irbis64r_91/cgiirbis_64.exe?P21DBN=KATL&I21DBN=KATL_PRINT&S21FMT=fullw_print&C21COM=F&Z21MFN=13707 (31 экземпляр)

б) дополнительная литература

1. Программирование на языке С / С. Кочан; пер. с англ. и ред. Г. В. Галисеева. - 3-е изд. - М. : Вильямс, 2007. - 496 с. http://kleopatra.pnzgu.ru/cgi-bin/irbis64r_91/cgiirbis_64.exe?P21DBN=KATL&I21DBN=KATL_PRINT&S21FMT=fullw_print&C21COM=F&Z21MFN=8079 (5 экземпляров)

в) программное обеспечение

Microsoft VISUAL STUDIO 2010. Договор № СД-130712001 от 12.07.2013 (подписка с 1 сентября 2013 г. до 31 августа 2017 г.). Продление Microsoft Imagine Standard KDF-00031 (подписка с 1 сентября 2017 г. до 31 августа 2020 г.)).

г) базы данных, информационно-справочные и поисковые системы Справка MSDN (www.microsoft.ru).

8. Материально-техническое обеспечение дисциплины (модуля)

Занятия по дисциплине «Программирование для ЭВМ» проводятся в лекционных и компьютерных аудиториях университета.

Рабочая программа дисциплины «Программирование для ЭВМ» составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 01.03.04 — «Прикладная математика».

Программу составили:

Тарасов Д. В., доцент кафедры «ВиПМ» _____


(Ф.И.О., должность, подпись)

Настоящая программа не может быть воспроизведена ни в какой форме без предварительного письменного разрешения кафедры-разработчика программы.

Программа одобрена на заседании кафедры «Высшая и прикладная математика»

Протокол № 7.1

от « 29 » 05 2015 года

Зав. кафедрой «ВиПМ»


_____ И. В. Бойков
(подпись, Ф.И.О.)

Программа согласована с заведующим выпускающей кафедрой
«Высшая и прикладная математика»

(название кафедры)



_____ И. В. Бойков
(подпись, Ф.И.О., дата)

Программа одобрена методической комиссией факультета вычислительной техники



Протокол № 6

от « 15 » июня 2015 года

Председатель методической комиссии
факультета вычислительной техники


_____ Н. Н. Коннов
(подпись) (Ф.И.О.)

Сведения о переутверждении программы на очередной учебный год и регистрации изменений

Учебный год	Решение кафедры (№ протокола, дата, подпись зав. кафедрой)	Внесенные изменения	Номера листов (страниц)		
			заменен- ных	новых	аннулированных
2016- 2017	№ 1, 19.09.2016 	Обновлен список литературы			
2017- 2018	№1, 04.09.2017 	Обновлен список литературы			