

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

УТВЕРЖДАЮ
Декан ФВТ

Л.Р. Фионова
« 11 » *сентября* 2016 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

С1.1.8 Программирование

Специальность: 09.05.01 «*Применение и эксплуатация автоматизированных систем специального назначения*»

Специализация №12: «*Автоматизированные системы обработки информации и управления специального назначения*»

Квалификация (степень) выпускника: *инженер*

Форма обучения: *очная*

Пенза, 2016

Рабочая программа дисциплины «Программирование» составлена в соответствии с требованиями ФГОС ВО по специальности 09.05.01 – «Применение и эксплуатация автоматизированных систем специального назначения».

Программу составил:

Убиенных Г.Ф., доцент кафедры «Информационно-вычислительные системы»



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

Программа одобрена на заседании кафедры «Информационно-вычислительные системы»

Протокол № 1

от « 06 » _____ 09 _____ 2016 г.

Зав. кафедрой ИВС



Косников Ю.Н.

Программа одобрена методической комиссией факультета ВТ

Протокол № 1

от « 22 » _____ 09 _____ 2016 г.

Председатель методической комиссии
факультета ВТ



Глотова Т.В.

Цели освоения дисциплины

Целями освоения дисциплины «**Программирование**» являются: подготовка студента к самостоятельному выполнению работ по созданию автоматизированных систем обработки информации и управления специального назначения, соответствующих общекультурным, общепрофессиональным, профессиональным и профессионально-специализированным компетенциям; овладение студентами знаниями и навыками в области разработки приложений.

2. Место дисциплины в структуре ОПОП специалитета

Учебная дисциплина «**Программирование**» относится к базовой части блока С1, шифр дисциплины С1.1.8.

Дисциплина опирается на знания, полученные студентами в ходе изучения дисциплин «Математика» и «Информатика».

Компетенции, приобретенные в ходе изучения дисциплины «**Программирование**», готовят студента к освоению профессиональных компетенций.

Знания, умения и навыки, полученные в результате изучения дисциплины, найдут применение при изучении следующих дисциплин: «Программирование на языке Java», «Технологии программирования», «Операционные системы», «Системное программное обеспечение», «Программирование в компьютерных сетях», а также при выполнении курсовых и дипломного проектов.

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

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

Коды компетенции	Наименование компетенции	Структурные элементы компетенции (в результате освоения дисциплины обучающийся должен знать, уметь, владеть)
ОПК-4	способность использовать языки и системы программирования, программные средства общего назначения, инструментальные средства компьютерного моделирования для решения различных исследовательских и профессиональных задач	Знать: методы структурного и объектно-ориентированного программирования
		Уметь: разрабатывать и отлаживать эффективные алгоритмы с использованием современных технологий программирования
		Владеть: навыками программирования в современных средах

4. Структура и содержание дисциплины «Программирование»

4.1. Структура дисциплины

Общая трудоемкость дисциплины составляет 12 зачетных единиц, 432 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				Защита лабораторных работ	Курсовая работа
				Всего	Лекции	Лабораторные занятия	Всего	Подготовка к лабораторным занятиям	Курсовая работа	Подготовка к экзамену		
1	Введение	2	1	1	1							
2	Раздел 1. Основы структурного программирования на языках C/C++	2	1-17	101	33	68	114	78		36		
3	Тема 1.1. Переменные и базовые типы языка	2	1	1	1		1			1		
4	Тема 1.2. Принятие решений. Условные операторы	2	2	2	2		1			1		
5	Тема 1.3. Организация циклов	2	3	2	2		1			1		
6	Тема 1.4. Числовые массивы	2	4-5	16	4	12	19	13		6	6	
7	Тема 1.5. Символьные массивы. Работа со строками	2	6-7	16	4	12	19	13		6	8	
8	Тема 1.6. Указатели	2	8	2	2		1			1		
9	Тема 1.7. Динамическое распределение памяти	2	9	14	2	12	19	13		6	10	
10	Тема 1.8. Функции	2	10-12	18	6	12	19	13		6	13	

4.2. Содержание дисциплины

4.2.1. Содержание лекционного курса

Введение

Краткая история возникновения языков C и C++. Цель и задачи дисциплины и ее место в подготовке специалиста.

1. Основы структурного программирования на языках C/C++

1.1. Переменные и базовые типы языка

Переменные. Константы. Базовые типы данных. Операции языка. Выражения. Преобразование типов в выражениях. Приведение типов.

1.2. Принятие решений. Условные операторы.

Операторы **if**, **if – else**, **if – else if – else**, **switch – case – default**, оператор условия **?**, операторы перехода **break**, **continue**, безусловный оператор перехода **goto**. Вложенные условные операторы. Логические условия.

1.3. Организация циклов

Операторы цикла **while**, **for**, **do – while**. Составные операторы цикла и операторы отношения. Оператор **continue**.

1.4. Числовые массивы

Одномерные массивы, их объявление. Двумерные массивы. Многомерные массивы. Инициализация массивов.

1.5. Символьные массивы. Работа со строками

Задание и инициализация символьных массивов. Библиотечные функции для работы со строками.

1.6. Указатели

Указатель как средство доступа к данным. Использование указателей в элементарных задачах программирования. Взаимосвязь указателей и массивов. Допустимые операции с указателями и массивами. Массивы указателей. Указатели на указатели.

1.7. Динамическое распределение памяти

Динамически распределяемая область памяти («куча»). Библиотечные функции **calloc()**, **malloc()**, **realloc()** и **free()**. Операторы **new** и **delete**.

1.8. Функции

Объявления и определений функций. Способы задания формальных параметров и типов возвращаемых данных. Вызов функций, передача аргументов по значению и ссылке. Функции с переменным числом аргументов. Подставляемые (встраиваемые) функции. Перегрузка функций. Рекурсивные функции.

1.9. Файловый ввод-вывод

Двоичные и текстовые потоки. Библиотечные функции для работы с файлами. Обработка файлов в потоковом режиме. Прямой доступ к данным файла.

1.10. Структуры и объединения

Определение структуры. Инициализация структуры. Массив структур. Оператор **typedef**. Структуры и функции. Объединения. Перечислимые типы.

1.11. Препроцессор языка

Назначение препроцессора. Основные директивы препроцессора.

1.12. Основы программирования в среде Embarcadero (CodeGear) C++ Builder.

Интерфейс среды. Библиотека визуальных компонентов. Инспектор объектов. Редактор кода. Обработчики событий. Работа с графикой.

2. Основы объектно-ориентированного и визуального программирования на языке C++

2.1. Классы.

Понятие абстрактных типов данных. Общедоступные, защищенные и скрытые данные. Область видимости класса.

Функции-члены классов. Инлайн-функции. Инициализация данных. Конструкторы и деструкторы.

Статические члены класса. Указатель this. Указатели на члены класса.

Дружественные функции и дружественные классы.

2.2. Наследование.

Построение производного класса.

2.3. Полиморфизм

Раннее и позднее связывание. Виртуальные функции. Абстрактные классы.

2.4. Шаблоны классов

Создание и использование шаблонов классов. Специализация шаблонов. Достоинства и недостатки шаблонов.

2.5. Обработка исключительных ситуаций

Общий механизм обработки исключений. Синтаксис исключений. Перехват исключений. Исключения в конструкторах и деструкторах. Иерархии исключений.

2.6. Стандартная библиотека классов

Потоковые классы. Строковый класс. Контейнерные классы. Итераторы. Алгоритмы. Математические классы.

2.7. Графика в C++

Использование графики в визуальных приложениях. Графический интерфейс. Рисование графически примитивов: точек, линий, прямоугольников, эллипсов. Отображение графиков функций с помощью графических примитивов.

2.8. Стандарт C++ 11

Изменения в классах. Лямбда-функции. Поддержка многопоточности.

Заключение

Основные тенденции развития языка C++.

4.2.2. Перечень и содержание лабораторных занятий

Основные цели лабораторного практикума – привитие студентам навыков в проектировании программ, в пользовании справочной литературой и справочными электронными ресурсами, а также подготовке их к самостоятельной работе над курсовым проектом. Тематика задач, решаемых на лабораторных занятиях, определяется содержанием соответствующих разделов рабочей программы. Содержание задач увязывается со спецификой будущей специальности. Форма проведения лабораторного практикума – решение задач, связанных с выполнением отдельных этапов проектирования программного продукта.

№ п/п	№ раздела	Наименование лабораторных работ	Кол. часов
1	1	Разработка программы по обработке числовых массивов (среда Microsoft Visual C++ – консольное приложение)	12
2	1	Разработка программы по обработке текстовой информации (среда Microsoft Visual C++ – консольное приложение)	12
3	1	Разработка приложения, использующего динамические массивы и указатели (среда Microsoft Visual C++ – консольное приложение)	12
4	1	Разработка программы с собственными функциями (среда Microsoft Visual C++ – консольное приложение)	12
5	1	Разработка приложения, использующего структуры и файлы (среда Microsoft Visual C++ – консольное приложение)	12
6	1	Разработка приложения по работе с графикой (среда Embarcadero (CodeGear) C++ Builder)	12

7	2	Разработка приложения, работающего с объектами классов (среда Microsoft Visual C++ – приложение Windows Forms)	10
8	2	Разработка приложения, использующего иерархию классов (среда Microsoft Visual C++ – приложение Windows Forms)	10
9	2	Разработка приложения по работе с шаблонами классов (среда Microsoft Visual C++ – приложение Windows Forms)	8
10	2	Разработка приложения по работе с файлами (среда Microsoft Visual C++ – приложение Windows Forms)	8
11	2	Разработка приложения по работе с контейнерными классами (среда Microsoft Visual C++ – приложение Windows Forms)	10
12	2	Разработка приложения по работе с графикой (среда Microsoft Visual C++ – приложение Windows Forms)	8

4.2.3. Содержание курсового проектирования

Курсовое проектирование заключается в разработке и отладке многомодульного приложения. Разработка ведется в среде Microsoft Visual C++ или Embarcadero Delphi & C++ Builder.

Темы курсовых работ

1. Медианный фильтр
2. Справочная система “Маршруты городского транспорта”
3. Программа тестирования студентов
4. Телефонный справочник
5. Расписание занятий студентов и преподавателей
6. Редактор схем алгоритмов
7. Мастер создания визитных карточек
8. Переводчик текстов
9. Игра «Пятнашки»
10. Электронные часы
11. Вычисление определенного интеграла
12. Решение нелинейного алгебраического уравнения
13. Клавиатурный тренажер
14. Учебный графический редактор
15. Органайзер
16. Решение системы линейных алгебраических уравнений
17. Решение нелинейного алгебраического уравнения
18. Численное решение обыкновенных дифференциальных уравнений
19. Системный будильник
20. Имитатор инсталлятора приложений
21. Имитатор почтовой системы
22. “Фоторобот”
23. Работа с комплексными числами
24. Работа с векторами
25. Работа с шестнадцатеричными данными
26. Шаблонный класс для разреженных массивов.
27. Инженерный калькулятор
28. Сотовый телефон
29. Имитатор работы ЛВС
30. Имитатор дефрагментатора жестких дисков

5. Образовательные технологии

В процессе изучения дисциплины применяются следующие образовательные технологии:

– чтение лекций проводится с использованием мультимедийного компьютерного проектора;

– практически все лекции и часть лабораторных занятий проводятся в интерактивной форме с разбором конкретных фрагментов программ, что составляет примерно 30 – 40 % аудиторных занятий;

– мастер-классы по работе в различных средах программирования (MS Visual C++, Embarcadero (CodeGear) C++ Builder);

– при выполнении лабораторного практикума и во время самостоятельной работы используются обучающие программы с сайта кафедры ИВС (<http://ivs-pgy.nm.ru>) и Интернет-ресурсы.

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

2-й семестр

№ нед.	Тема	Вид самостоятельной работы	Задание	Рекомендуемая литература	Кол-во часов
5	Основы структурного программирования на языках C/C++	Подготовка к лабораторному занятию. Оформление отчета по лабораторной работе	Изучение работы с одномерными и двумерными числовыми массивами. Разработка программы по обработке числовых массивов	/1, 2/	12
7	Основы структурного программирования на языках C/C++	Подготовка к лабораторному занятию. Оформление отчета по лабораторной работе	Изучение библиотечных функций для работы со строками. Разработка программы по обработке текстовой информации	/1, 2/	12
9	Основы структурного программирования на языках C/C++	Подготовка к лабораторному занятию. Оформление отчета по лабораторной работе	Изучение работы операторов new и delete . Разработка приложения, использующего динамические массивы и указатели		12
	Основы структурного про-	Подготовка к лабораторному занятию.	Изучение объявления, определения и вызова		

12	граммирования на языках C/C++	Оформление отчета по лабораторной работе	функций. Разработка программы с собственными функциями	/1, 2/	12
14	Основы структурного программирования на языках C/C++	Подготовка к лабораторному занятию. Оформление отчета по лабораторной работе	Изучение библиотечных функций по работе с файлами. Разработка приложения, использующего структуры и файлы	/1, 2/	12
17	Основы структурного программирования на языках C/C++	Подготовка к лабораторному занятию. Оформление отчета по лабораторной работе	Изучение интерфейса среды Embarcadero (CodeGear) C++ Builder и ее графических возможностей. Разработка приложения по работе с графикой	/1, 2/	12

3-й семестр

№ нед.	Тема	Вид самостоятельной работы	Задание	Рекомендуемая литература	Кол-во часов
3	Основы объектно-ориентированного и визуального программирования на языке C++	Подготовка к лабораторному занятию. Оформление отчета по лабораторной работе	Изучение структуры класса и ее программной реализации. Разработка приложения, работающего с объектами классов	/1, 2/	6
5	Основы объектно-ориентированного и визуального программирования на языке C++	Подготовка к лабораторному занятию. Оформление отчета по лабораторной работе	Изучение механизмов наследования и полиморфизма. Разработка приложения, использующего иерархию классов	/1, 2/	6
8	Основы объектно-ориентированного и визуального программирования на языке C++	Подготовка к лабораторному занятию. Оформление отчета по лабораторной работе	Изучение механизма создания шаблонов классов. Разработка приложения по работе с шаблонами классов	/1, 2/	6
12	Основы объектно-ориентированного и визу-	Подготовка к лабораторному занятию.	Изучение механизмов библиотечных файловых		

	ального программирования на языке C++	Оформление отчета по лабораторной работе	потоков. Разработка приложения по работе с файлами	/1, 2/	8
14	Основы объектно-ориентированного и визуального программирования на языке C++	Подготовка к лабораторному занятию. Оформление отчета по лабораторной работе	Изучение библиотечных контейнерных классов. Разработка приложения по работе с контейнерами	/1, 2/	8
15	Основы объектно-ориентированного и визуального программирования на языке C++	Подготовка к лабораторному занятию. Оформление отчета по лабораторной работе	Изучение библиотечных функций по работе с графикой. Разработка приложения по работе с графикой	/1, 2/	6

6.2. Методические указания по организации самостоятельной работы студентов

Планируются следующие виды самостоятельной работы:

- подготовка к лабораторным работам занятиям;
- оформление отчётов по лабораторным работам;
- работа по выполнению курсовой работы;
- подготовка к зачету;
- работа с конспектом лекций и изучение литературы при подготовке к экзаменам.

В качестве учебно-методического обеспечения по организации самостоятельной работы студентов при подготовке к аудиторным занятиям рекомендуется использовать основную и дополнительную литературу, а также электронные методические указания по данной дисциплине, размещенные в университетской локальной сети. Также во время самостоятельной работы используются материалы сайта «Интернет Университет Информационных Технологий» (<http://www.intuit.ru>).

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
1	Текущий: собеседование при защите лабораторных работ	Разделы 1 – 2	ОПК-4
2	Промежуточный: защита курсовой работы	Разделы 1 – 2	ОПК-4
3	Промежуточный: зачет по результатам защиты лабораторных работ	Разделы 1 – 2	ОПК-4
4	Промежуточный: экзамен (вопрос и задание)	Разделы 1 – 2	ОПК-4

Примерный перечень вопросов и заданий к экзамену

Вопросы к экзамену

1. Структура программы на языке C/C++.
2. Понятия константы и переменной.
3. Статические и динамические переменные.
4. Глобальные и локальные данные.
5. Базовые типы данных языка C/C++: объявления, область значения, константы.
6. Логические тип данных в C++.
7. Стандартные целочисленные типы данных языка C/C++.
8. Представление вещественных чисел. Стандартные вещественные типы языка C/C++.
9. Представление символьных данных. Стандартный тип CHAR.
10. Что называется массивом в языках программирования?
11. Что называется структурой (записью) в языках программирования?
12. Логические операции в языках программирования.
13. Каким образом реализуется простое ветвление в программе?
14. Каким образом реализуется множественное ветвление в программе?
15. Каким образом реализуется цикл с предусловием в программе?
16. Каким образом реализуется цикл с постусловием в программе?
17. Каким образом реализуется цикл с заданным числом повторений?
18. Что такое составной оператор и как он реализуется?
19. Операторы досрочного завершения цикла.
20. Для чего и как выполняется связывание физического файла с логическим файлом?
21. Перечислите основные файловые типы.
22. Чем отличается последовательный доступ к компонентам файла от прямого?
23. Что такое указатель в программе?
24. Что такое разыменованье?
25. Что такое линейный односвязный список?
26. Принцип структурного программирования.
27. Чем функция main() отличается от других функций в C/C++?
28. Что такое рекурсия?
29. Структура модуля.
30. Концепции восходящего и нисходящего программирования.
31. Понятия объекта и класса в объектно-ориентированном программировании.
32. Принцип инкапсуляции в объектно-ориентированном программировании.
33. Наследование в объектно-ориентированном программировании.
34. Раннее и позднее связывание.
35. Понятие виртуальных методов в объектно-ориентированном программировании.
36. Какой метод класса называется конструктором?
37. Какой метод класса называется деструктором?
38. Понятие компонента и принцип компонентного программирования.
39. Что называется свойством компонента?
40. Понятие события компонента. Обработчик события.

Задания к экзамену

Среда разработки – Microsoft Visual C++ (консольное приложение).

1. Разработать программу, которая определяет, является ли введенная последовательность целых чисел произвольной длины монотонно возрастающей или нет.

2. Разработать программу, которая вводит текст на английском языке и выводит в алфавитном порядке все буквы, которые входят в этот текст по одному разу.

3. Разработать программу, которая вводит текст произвольной длины, определяет и выводит частоту распределения длин слов текста. (Знаками препинания в тексте пренебречь).

4. Разработать программу, которая вводит текст и выводит самое длинное слово (слова) этого текста.

5. Разработать программу, которая вводит количество точек, расположенных на плоскости, и находит пару самых удаленных друг от друга точек. Координаты точек формируются программой случайным образом.

6. Дан одномерный целочисленный массив из N элементов (N вводится), значения которых формируются случайным образом в диапазоне $[-99, 99]$. Найти количество и процентное соотношение положительных, отрицательных и нулевых элементов.

7. Разработать и испытать функцию для проверки баланса парных скобок в строке, то есть количество открытых скобок должно совпадать с количеством закрытых, причём необходимо отличать тип скобок – круглые, квадратные, фигурные. Функция возвращает значение типа `int` равное **0** – если баланс скобок нарушен, равное **1** – если баланс скобок соблюдается, равное **-1** – если скобок в строке нет.

```
// прототип функции проверки баланса парных скобок  
int balanceBrackets(char *);
```

8. Дан одномерный целочисленный массив из N элементов, заданных случайными числами на промежутке $[a; b)$. Значения N, a, b вводятся с клавиатуры. Заменить все элементы массива, кратные 3, на сумму их цифр.

9. Дан одномерный целочисленный массив из N элементов, заданных случайными числами на промежутке $[a; b)$. Значения N, a, b вводятся с клавиатуры. Поменять местами первый минимальный и последний максимальный элементы массива.

10. Разработать программу выбора из заданного на плоскости множества точек N (N вводится) трех точек, не лежащих на одной прямой, составляющих треугольник наименьшей площади. Координаты точек формируются программой случайным образом.

11. Заданы два вектора с различным количеством элементов и натуральное число k (количество элементов векторов и k вводятся с клавиатуры). Объединить их в один вектор, включив второй вектор между k -м и $(k+1)$ -м элементами первого. Векторы должны храниться в памяти в виде динамических массивов, элементы их – случайные целые числа, распределенные по равномерному закону на интервале $[-99; 99]$.

12. Разработать программу, определяющую сумму накопления в банке по первоначальному вкладу, виду счета и количеству прошедших лет.

13. Разработать программу, вычисляющую $e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!} + \dots$ с точностью ϵ (значения x и ϵ вводятся с клавиатуры).

14. Известно, что в целочисленной матрице нет нулей. Заменить повторяющиеся в матрице элементы нулями. Элементы матрицы генерируются случайным образом в диапазоне $[-99, 99]$, размеры матрицы вводятся пользователем.

15. Разработать программу поиска максимального элемента среди минимальных элементов строк целочисленной матрицы размером $n \times m$. Значения элементов матрицы формируются случайным образом в диапазоне $[a, b]$ (значения n, m, a, b вводятся).

16. Разработать программу, которая формирует матрицу размером $m \times n$ (m и n вводятся с клавиатуры, элементы матрицы – случайные целые числа, распределенные по равномерному закону на интервале $[-99, 99]$), находит среднее арифметическое элементов матрицы, расположенных на главной диагонали и увеличивает каждый элемент матрицы на эту величину.

17. Разработать и испытать функцию "сжатия" исходной последовательности символов: каждая подпоследовательность, состоящая из нескольких вхождений одного и того

же символа заменяется на текст $x(k)$, где x – символ, а k – строка, являющаяся записью числа вхождений в исходную последовательность.

18. Разработать программу, которая в вещественной квадратной матрице сортирует по возрастанию элементы каждой строки верхнего треугольника, по убыванию – нижнего, оставляет без изменения диагональные элементы. Элементы матрицы генерируются случайным образом в диапазоне $[-99.99, 99.99]$, размер матрицы вводится с клавиатуры.

19. Разработать и испытать функцию сложения двух простых дробей a/b и c/d . В результате должна получаться правильная несократимая дробь или смешанное число.

20. Разработать программу, которая в целочисленной квадратной матрице находит максимальный элемент(ы), находящийся на периметре матрицы, и минимальный элемент(ы) вне периметра матрицы, и выводит их вместе с индексами. Значения элементов матрицы генерируются случайным образом в диапазоне $[a, b]$, размер матрицы и значения a, b вводятся пользователем.

21. Разработать программу, которая формирует целочисленный динамический массив, удаляет в нем наибольший и наименьший элементы и находит в преобразованном массиве среднее арифметическое k наибольших элементов. Значения элементов массива генерируются случайным образом в диапазоне $[a, b]$, размер массива и значения a, b, k вводятся пользователем.

22. Разработать программу, которая редактирует текст, введенный пользователем, удаляя все слова с нечетными номерами и переворачивая слова с четными номерами. Нумерация слов текста начинается с единицы.

23. Разработать программу, которая в целочисленной матрице находит строки с наибольшей и наименьшей суммой значений их элементов и выводит найденные строки и суммы их элементов. Поиск строк с наибольшей и наименьшей суммой значений их элементов оформить в виде функций. Значения элементов матрицы генерируются случайным образом в диапазоне $[a, b]$, размеры матрицы и значения a, b вводятся пользователем.

24. Разработать программу, которая из квадратной целочисленной матрицы формирует вектор, элементами которого являются суммы элементов тех строк матрицы, которые начинаются с k идущих подряд положительных чисел. Значения элементов матрицы генерируются случайным образом в диапазоне $[-99, 99]$, размер матрицы и значение k вводятся пользователем.

7. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература

1. Конова, Е.А. Алгоритмы и программы. Язык C++: учеб. пособие / Е.А. Конова, Г.А. Поллак. – СПб.: Лань, 2016. – 384 с. [Электронный ресурс]. Режим доступа: <https://e.lanbook.com/reader/book/90158>

2. Павловская, Т.А. C/C++. Процедурное и объектно-ориентированное программирование: учеб. пособие / Т.А. Павловская. – СПб.: Питер, 2015. – 495 с.

б) дополнительная литература

3. Керниган, Б. У. Язык программирования Си / Б. У. Керниган, Д. М. Ритчи. – 2-е изд. – М.: Вильямс, 2013. – 304 с.

4. Александров, Э. Э. Программирование на языке C в Microsoft Visual Studio 2010: учеб. пособие / Э. Э. Александров, В. В. Афонин. – Саранск: Изд-во Мордов. ун-та, 2010. – 428 с.

5. Иванова, Г.С. Объектно-ориентированное программирование: учебник для вузов. – 3-е изд. / Г.С. Иванова, Т.Н. Ничушкина, Е.К. Пугачев / Под ред. Г.С. Ивановой. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2007. – 368 с.

6. Подбельский, В. В. Программирование на языке Си: учеб. пособие / В. В. Подбельский, С. С. Фомин. – 2-е изд. – М.: Финансы и статистика, 2007. – 600 с.

7. Павловская, Т.А. С++. Объектно-ориентированное программирование: Практикум / Павловская Т.А., Щупак Ю.А. – СПб.: Питер, 2006. – 265 с. – (Учеб. Пособие).
8. Страуструп Б. Язык программирования С++. Специальное издание. – М.: «Бином – Пресс», 2008. – 1104 с.
9. Холзнер С. Visual С++ 6: Учебный курс. – СПб.: Питер, 2008. – 570 с.
10. Прата, С. Язык программирования С++. Лекции и упражнения, 6-е изд. – М.: ООО «И.Д. Вильямс», 2012. – 1248 с.
11. Кент, Д. С++. Основы программирования. – М.: НТ Пресс, 2008. – 366 с. (Самочуитель).
12. Довбуш, Г.Ф. Visual С++ на примерах/ Г.Ф. Довбуш, А.Д. Хомоненко/ под ред. проф. А.Д. Хомоненко. – СПб.: БХВ – Петербург, 2007. – 528 с.

в) программное обеспечение и Интернет – ресурсы

1) Программные продукты Microsoft Visual С++ версии 2010 или выше, Embarcadero (CodeGear) С++ Builder версии 2009 или выше.

2) Интернет-ресурсы

– Материалы раздела «Программирование» сайта «Интернет – Университет Информационных Технологий», <http://www.intuit.ru/>

– Липачев Е.К. Технология программирования. Базовые конструкции С/С++: учебно-справочное пособие. – Казань: Казан. ун-т, 2012. – 142 с. (http://window.edu.ru/resource/139/78139/files/prog_C.pdf).

– Александров Э.Э., Афонин В.В. Программирование на языке С в Microsoft Visual Studio 2010 : учеб. пособие. – Саранск: Изд-во Мордов. ун-та, 2010. – 428 с. (Интернет-Университет Информационных Технологий (INTUIT.ru) – <http://www.intuit.ru/department/pl/prcmsvs2010/>).

– Алексеева Н.Н. Основы программирования. Типы данных: учебное пособие по курсу «Основы программирования» (презентация). – Сайт Санкт-Петербургского колледжа информационных технологий

(http://www.spbkit.edu.ru/index.php?option=com_content&task=view&id=44&Itemid=112/cpp2.pps).

– Шапкина Л.М. Работа с объектами в С++ (презентация). – Сайт Санкт-Петербургского колледжа информационных технологий.

(http://www.spbkit.edu.ru/index.php?option=com_content&task=view&id=44&Itemid=112/sha_1.pps).

– Глухих М.И. Основы программирования на языках С и С++: Курс лекций «Программирование» (презентация). – Сайт кафедры компьютерных систем и программных технологий СПбГПУ (<http://kspt.ftk.spbstu.ru/course/cpp14>).

8. Материально-техническое обеспечение дисциплины

Лекционные занятия проводятся в аудитории, оснащенной компьютерным проектором, проекционным экраном, шторами, сетью электропитания 220 В.

Лабораторные занятия проводятся в классе, оснащенный персональными компьютерами с операционной системой Windows XP/Windows Vista/Windows 7/8/10 или Linux.

