

Учреждение образования «Полоцкий государственный университет»

УТВЕРЖДАЮ

Проректор по учебной работе
учреждения образования
«Полоцкий государственный
университет»

 Ю.П. Голубев

« 30 » 2021 г.

Регистрационный №УД- 178/21уч

Модуль «Программирование»

ОСНОВЫ И МЕТОДОЛОГИИ ПРОГРАММИРОВАНИЯ

Учебная программа учреждения высшего образования
по учебной дисциплине для специальности
1-98 01 01 Компьютерная безопасность
(по направлениям)
направление специальности
1-98 01 01-01 Компьютерная безопасность
(математические методы и программные системы)
специализация 1-98 01 01-01 03 Защищенные информационные системы

2021 г.

Учебная программа составлена на основе учебного плана специальности 1-98 01 01 «Компьютерная безопасность (по направлениям)». Регистрационный №21-21/уч. ФКНиЭ от 26.07.2021 г. для дневной очной формы получения высшего образования.

СОСТАВИТЕЛЬ:

Ирина Брониславовна Бураченко, доцент кафедры математики и компьютерной безопасности учреждения образования «Полоцкий государственный университет», к.т.н.

РЕЦЕНЗЕНТЫ:

К.Я. Раханов (к.т.н., доцент, технический директор ООО «ТриИнком»)
В.О. Козлов (Lead Software at .Net Division ООО «Godel Technologies Europe»)

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой математики и компьютерной безопасности учреждения образования «Полоцкий государственный университет»
(протокол № 9 от «20» 09 2021 г.).

Методической комиссией факультета компьютерных наук и электроники учреждения образования «Полоцкий государственный университет»
(протокол № 1 от «21» 09 2021 г.).

Научно-методическим советом учреждения образования «Полоцкий государственный университет»
(протокол № 1 от «30» 09 2021 г.).

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Учебная дисциплина «Основы и методологии программирования» формирует и развивает у студентов алгоритмическое мышление и способствует их обучению методам решения разнообразных задач с использованием персональных компьютеров и современных систем программирования.

Целью изучения дисциплины «Основы и методологии программирования» является формирование у студентов теоретических знаний и практических навыков решения различных типов задач с использованием методологий программирования. Изучение данной дисциплины является необходимым этапом в профессиональном развитии «специалиста по защите информации. математика».

Задачи изучения учебной дисциплины «Основы и методологии программирования». При изучении данной дисциплины требуется разрешить основные задачи:

- развитие математического, логико-алгоритмического и программистского стилей мышления;
- ознакомление с основами алгоритмизации и высокоуровневыми алгоритмическими языками программирования;
- формирование практических знаний и умений использования современных методологий, методов и систем программирования;
- обучение основам разработки программного обеспечения;
- изучение основ технологии проектирования программ;
- формирование представления о видах и типах языков программирования и других искусственных языков, необходимых для систематизации и хранения данных для программ;
- обучение объектно-ориентированному языку C++, позволяющему осваивать классические приемы и современные технологии программирования;
- формирование у студентов представления о современных подходах и методах программного решения научных и прикладных задач.

В результате изучения учебной дисциплины студент должен:

знать:

- методы решения научно-технических и информационных задач;
- основные высокоуровневые алгоритмические языки программирования;

уметь:

- решать типовые задачи математики и информатики;
- работать на современных вычислительных средствах;
- описывать алгоритмы программ;
- применять современные информационные технологии и методы реализации решения прикладных задач на высокоуровневых языках программирования;

владеть:

- методами программирования задач в различных областях;
- современными технологиями разработки программ;
- основами алгоритмизации и программирования на одном или более языков программирования высокого уровня.

иметь представление:

- о структуре процесса разработки программных средств.

В качестве базового учебного языка программирования выбраны языки Pascal и C. Полученные навыки далее развиваются посредством обучения объектно-ориентированному языку C++, позволяющему осваивать классические приемы и современные технологии программирования.

Требования к уровню освоения содержания учебной дисциплины. При изучении дисциплины «Основы и методологии программирования» у студентов специальности специальности 1-98 01 01 «Компьютерная безопасность (по направлениям)» должен сформироваться набор компетенций связанных с программированием, соответствующих присваиваемой по завершению высшего образования квалификации «Специалист по защите информации. Математик» обеспечивающих выпускникам по указанной специальности

успешность применения полученных знаний и умений в дальнейшей профессиональной деятельности:

Базовые профессиональные компетенции:

БПК-2 Строить, анализировать и тестировать алгоритмы и программы решения типовых задач обработки информации с использованием структурного, объектно-ориентированного и иных парадигм программирования.

Сформированные компетенции являются базовыми при изучении всех последующих дисциплин, связанных с программированием, а также фундаментальной основой для дальнейшей профессиональной деятельности специалиста в области защиты информации.

Перечень дисциплин, в продолжение и на базе которых изучается дисциплина.

Для изучения учебной дисциплины «Основы и методологии программирования» по специальности 1-98 01 01 «Компьютерная безопасность (по направлениям)» студенты опираются на знания, полученные в области общего среднего образования по информатике, математике и физике, также необходимы базовые знания и навыки работы с персональным компьютером.

Перечень дисциплин, которые изучаются на базе дисциплины.

Знания полученные при изучении дисциплины «Основы и методологии программирования» по специальности 1-98 01 01 «Компьютерная безопасность (по направлениям)» являются основой для дисциплин: «Промышленное программирование», «Разработка кросс-платформенных приложений», «Машинно-ориентированное программирование», «Технологии программирования», «Алгоритмы и структуры данных» и «Базы данных», «Модели данных и системы управления базами данных» и пр. Изучение учебной дисциплины позволяет дать студентам базу, необходимую для успешного усвоения материала перечисленных выше учебных дисциплин, а также получить знания, необходимые им в дальнейшем для успешной работы по специальности.

В соответствии с учебным планом по специальности 1-98 01 01 «Компьютерная безопасность (по направлениям)» на изучение учебной дисциплины отводится:

Форма получения высшего образования первой ступени	дневная
Курс (курсы)	1
Семестр	1
Всего часов по дисциплине	216
Всего аудиторных часов по дисциплине	136
В том числе:	
Лекции, часов	68
Лабораторные занятия, часов	68
Самостоятельная работа, часов	80
Форма текущей аттестации	зачет, экзамен
Трудоёмкость дисциплины, зач. ед	6

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

ВВЕДЕНИЕ В ДИСЦИПЛИНУ

Цели и задачи изучения дисциплины. Содержание и структура дисциплины. Основные термины и определения, используемые в материале.

РАЗДЕЛ 1 ОСНОВЫ МЕТОДОЛОГИИ ПРОГРАММИРОВАНИЯ

Тема 1.1 Основы методологии программирования.

Стили и методы программирования. Основные определения и понятия: программа, программный продукт, программный комплекс, жизненный цикл программного обеспечения. Основные парадигмы программирования.

Тема 1.2 Императивное программирование.

Вычислительная модель, методы и концепции, абстрактная машина Тьюринга, синтаксис и семантика, процедурное программирование, императивные языки программирования, технология процедурного программирования, классы решаемых задач, проблемы классического программирования.

Тема 1.3 Модульно-процедурное программирование.

Назначение модулей, возникновение модуля, основные концепции модульного программирования: принцип утаивания информации Парнаса, аксиома модульности Коуэна, сборочное программирование Цейтина. Определение модуля и его примеры. Разновидности модулей. Набор характеристик модуля: размер модуля, прочность модуля, сцепление модулей, рутинность. Понятие процедуры. Основные понятия и языки модульного и процедурного программирования

Тема 1.4 Структурное программирование

Методология структурного императивного программирования. Происхождение, история и эволюция. Основные методы и концепции. Теорема Бёма-Джакопини. Основные понятия и языки структурного программирования.

Тема 1.5 Метод объектно-ориентированного программирования (ООП)

Метод объектной декомпозиции. Методы и концепции ООП: концепция инкапсуляции, концепция наследования, концепция полиморфизма. Вычислительная модель ООП. Синтаксис и семантика. Основные свойства ООП. Языки ООП.

Тема 1.6 Логическое программирование

Возникновение направления и исторические основы логического программирования. Недостатки логического программирования. Примеры программ на языке Prolog.

Тема 1.7 Функциональное программирование.

Основные идеи функционального подхода. Преимущества и недостатки. Понятие функции. Исчисление лямбда-конверсий. Эволюция языков функционального программирования.

Тема 1.8 Стили и методы программирования.

Традиционные языки программирования. Сравнение программ на разных языках. Работа программной системы над текстом программы. Модели вычислений и их различия. Особенности элементов традиционной архитектуры вычислительной системы. Модификации традиционной архитектуры. Стековая архитектура. RISC архитектура. Нетрадиционные архитектуры. Многопроцессорность.

РАЗДЕЛ 2 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

Тема 2.1 Алгоритмизация вычислительных процессов

Основные определения и понятия. Этапы разработки программ. Методы проектирования алгоритмов и программ. Принципы структурного программирования. Понятия алгоритма и его свойства. Способы отображения алгоритмов. Базовые алгоритмические структуры.

Тема 2.2 Основные элементы языков Pascal и C

Общая характеристика языка Pascal. Алфавит языка Pascal. Структура программы. Основные объекты программы. Типы данных и операции, производимые с ними. Стандартные процедуры и функции. Арифметические выражения. Оператор присваивания. Процедуры ввода-вывода данных. Метки и оператор безусловного перехода.

Тема 2.3 Ветвления. Массивы. Циклы

Простые и сложные условия. Операторы ветвления: if и case. Массивы: описание и использование. Операторы цикла: for, while и repeat. Прерывание циклов: break, continue и goto. Решение некоторых типовых задач.

Тема 2.4 Сортировка массивов

Постановка задачи сортировки. Простые алгоритмы сортировки: метод «пузырька», метод «пузырька» с флагом, метод выбора, сортировка простыми вставками, метод прямых вставок с барьером, сортировка бинарными вставками. Быстрые алгоритмы сортировки: сортировка Шелла, пирамидальная сортировка, «быстрая» сортировка. Алгоритмы поиска: линейный поиск, двоичный поиск, сравнение методов поиска.

Тема 2.5 Строки. Множества

Символьные и строковые константы. Стандартные функции и процедуры обработки строк. Примеры обработки символьных строк. Множества. Операции с множествами. Примеры использования символов, строк и множеств.

Тема 2.6 Подпрограммы в языке Pascal и C

Структура, назначение и применение подпрограмм. Параметры и аргументы, области действия имен. Процедуры. Функции. Рекурсивные процедуры и функции. Примеры решения задач.

Тема 2.7 Записи

Понятие записи. Объявление записи и обращение к полям записи. Оператор присоединения With. Массив записей. Записи в процедурах и функциях. Файлы записей. Чтение из файла. Примеры программ. Выделение памяти под запись. Сортировка массива записей.

Тема 2.8 Файлы

Общие сведения о файлах и файловой системе. Файлы языка Pascal. Текстовые файлы. Типизированные файлы. Примеры решения задач обработки файлов.

Тема 2.9 Динамические структуры данных

Адресация оперативной памяти MS DOS. Указатели и операции над ними. Управление динамической памятью. Работа с динамическими массивами. Динамические структуры данных: списки, стеки, очереди, деки. Деревья и графы. Примеры решения задач обработки динамических структур данных.

Тема 2.10 Модульное программирование

Модули. Открытые массивы и строки. Нетипизированные параметры. Параметры процедурного типа. Примеры решения задач с использованием модулей.

Тема 2.11 Приемы программирования типовых задач

Записи. Типизированные файлы. Текстовые файлы. Списки.

РАЗДЕЛ 3. ЯЗЫК ПРОГРАММИРОВАНИЯ C++. СТРУКТУРНОЕ ПРОГРАММИРОВАНИЕ

Тема 3.1 Базовые сведения о языке C++

Алфавит языка. Идентификаторы. Ключевые слова. Комментарии. Переменные. Константы и литералы. Преобразование типов.

Тема 3.2 Структура программы на C++

Блок заголовков программы. Блок с прототипами. Главная функция программы. Блок с описаниями функций. Примеры программ.

Тема 3.3 Вычисления в C++

Понятие операции и выражения в C++. Арифметические операции. Операция присваивания. Операции сравнения. Логические операции. Операция размера.

Тема 3.4 Управляющие инструкции

Условные операторы: оператор if, оператор switch. Операторы организации циклов: оператор for, оператор while, оператор do while.

Тема 3.5 Указатели

Понятие переменных-указателей. Операторы, используемые при работе с указателями. Арифметические действия с указателями. Стек и динамически распределяемая память. Указатели и объекты в динамической памяти.

Тема 3.6 Массивы

Одномерные массивы. Двумерные массивы. Инициализация одномерных массивов. Инициализация двумерных массивов. Массивы и указатели. Массивы, расположенные в динамической памяти. Массивы символов. Массив строк.

Тема 3.7 Структуры и объединения

Определение структуры. Определение структурной переменной. Доступ к полям структуры. Инициализация структурных переменных. Объединения.

Тема 3.8 Функции

Объявление, определение и вызов функции. Использование аргументов. Использование инструкции return. Версия инструкции return, которая не возвращает значение. Версия инструкции return, которая возвращает значение. Правила действия областей видимости. Локальная область видимости. Соккрытие имен переменных. Параметры функции. Глобальная область видимости. Перегрузка функций. Способы передачи аргументов в функции. Передача аргумента по значению. Передача аргумента по ссылке. Ссылочные параметры. Передача функции массива. Передача функциям строк. Возвращение функциями указателей и ссылок.

Тема 3.9 Функции и составные типы данных

Функции и структуры. Указатель на функцию. Указатели на тип void. Указатели на функции, аргументами или возвращаемыми значениями которых является указатель на void.

Учебно-методическая карта учебной дисциплины «Основы и методологии программирования»

Номер раздела, темы, занятия	Название раздела, темы, занятия; перечень изучаемых вопросов	Количество аудиторных часов				Литература	Формы контроля знаний
		лекции	Лабораторные занятия	Практические занятия	Управляемая самостоятельная работа студента		
1	2	3	4	5	6	7	8
	Введение в дисциплину	2	2				
1	Лекция № 1 <i>Введение в дисциплину</i> Цели и задачи изучения дисциплины. Содержание и структура дисциплины. Основные термины и определения, используемые в материале.	2				Осн. лит.: [1]. Доп. лит.: [1], [3]. Библ. online: [3].	
2	Лабораторная работа №1 <i>Введение в Pascal и C</i> Знакомство с интерфейсом программы PascalABC.NET. Базовые навыки написания программ на языке Pascal. Ввод и вывод данных с консоли. Простейшие операции.		2			Методические указания	Защита отчета по лабораторной работе № 1
	Раздел 1. Основы методологии программирования	20	20				
3	Лекция № 2 <i>Тема 1.1 Основы методологии программирования.</i> Стили и методы программирования. Основные определения и понятия: программа, программный продукт, программный комплекс, жизненный цикл программного обеспечения. Основные парадигмы программирования.	2				Осн. лит.: [1]. Доп. лит.: [1], [3], [4], [5], [6]. Библ. online: [3]. Эл. рес.: [1].	Блиц-опрос

1	2	3	4	5	6	7	8
4	<p>Лабораторная работа №2 <i>Выражения, их типы.</i> Алфавит языка. Арифметические выражения. Некоторые стандартные функции, типы аргумента и результаты. Операции отношений и логические операции. Приоритет операций.</p>		2			Методические указания	Защита отчета по лабораторной работе № 2
5	<p>Лекция № 3 <i>Тема 1.2 Императивное программирование.</i> Вычислительная модель, методы и концепции, абстрактная машина Тьюринга, синтаксис и семантика, процедурное программирование, императивные языки программирования, технология процедурного программирования, классы решаемых задач, проблемы классического программирования.</p>	2				Осн. лит.: [1]. Доп. лит.: [1], [3], [4]. Библ. online: [1], [3]. Эл. рес.: [1].	Блиц-опрос
6	<p>Лабораторная работа №3 <i>Программирование алгоритмов линейной структуры.</i> Окно редактора кода. Окно вывода. Окно ввода. Запуск и остановка программы. Структура программы. Константы и переменные. Целочисленные типы данных. Вещественный тип данных. Символьный тип данных. Логический тип данных. Процедуры ввода-вывода данных. Оператор присваивания. Пустой оператор.</p>		2			Методические указания	Защита отчета по лабораторной работе № 3
7	<p>Лекция № 4 <i>Тема 1.3 Модульно-процедурное программирование.</i> Назначение модулей, возникновение модуля, основные концепции модульного программирования: принцип утаивания информации Парнаса, аксиома модульности Коуэна, сборочное программирование Цейтина.</p>	2				Осн. лит.: [1]. Доп. лит.: [1], [3]. Библ. online: [3], [4]. Эл. рес.: [1].	Блиц-опрос
8	<p>Лабораторная работа №4 <i>Программирование алгоритмов разветвляющейся структуры.</i> Оператор условного перехода If. Оператор выбора Case. Составной оператор.</p>		2			Методические указания	Защита отчета по лабораторной работе № 4

1	2	3	4	5	6	7	8
9	<p>Лекция № 5 Определение модуля и его примеры. Разновидности модулей. Набор характеристик модуля: размер модуля, прочность модуля, сцепление модулей, рутинность. Понятие процедуры. Основные понятия и языки модульного и процедурного программирования.</p>	2				Осн. лит.: [1]. Доп. лит.: [1], [3]. Библ. online: [3], [4], [5]. Эл. рес.: [1].	Блиц-опрос
10	<p>Лабораторная работа №5 <i>Программирование алгоритмов циклической структуры.</i> Понятие тела цикла. Оператор цикла While. Оператор цикла Repeat. Оператор цикла For. Способы отображения на блок-схемах.</p>		2			Методические указания	Защита отчета по лабораторной работе № 5
11	<p>Лекция № 6 <i>Тема 1.4 Структурное программирование</i> Методология структурного императивного программирования. Происхождение, история и эволюция. Основные методы и концепции. Теорема Бёма-Джакопини. Основные понятия и языки структурного программирования.</p>	2				Осн. лит.: [1]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [3].	Блиц-опрос
12	<p>Лабораторная работа №6 <i>Программирование алгоритмов с использованием вложенных циклов.</i> Понятие вложенного цикла. Внешний и внутренний циклы. Правила организации вложенных циклов. Особенности отображения на блок-схемах.</p>		2			Методические указания	Защита отчета по лабораторной работе № 6
13	<p>Лекция №7 <i>Тема 1.5 Метод объектно-ориентированного программирования</i> Метод объектной декомпозиции. Методы и концепции ООП: концепция инкапсуляции, концепция наследования, концепция полиморфизма. Вычислительная модель ООП. Синтаксис и семантика. Основные свойства ООП. Языки ООП.</p>	2				Осн. лит.: [1]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [3].	Блиц-опрос
14	<p>Лабораторная работа №7 <i>Строковые переменные.</i> Данные строкового типа. Объявление строковой переменной. Строковые выражения. Операции присваивания, объединения, сравнения и отношения над строками. Стандартные функции для работы со строками. Стандартные процедуры для работы со строками.</p>		2			Методические указания	Защита отчета по лабораторной работе № 7

1	2	3	4	5	6	7	8
15	<p>Лекция № 8 <i>Тема 1.6 Логическое программирование</i></p> <p>Возникновение направления и исторические основы логического программирования. Недостатки логического программирования. Примеры программ на языке Prolog.</p>	2				<p>Осн. лит.: [1]. Доп. лит.: [1], [3].</p> <p>Библ. online: [3]. Эл. рес.: [3].</p>	Блиц-опрос
16	<p>Лабораторная работа №8 <i>Программирование с использованием пользовательских процедур и функций.</i></p> <p>Понятие подпрограммы. Понятие процедуры. Понятие функции. Встроенные и определённые пользователем процедуры и функции.</p>		2			Методические указания	Защита отчета по лабораторной работе № 8
17	<p>Лекция № 9 <i>Тема 1.7 Функциональное программирование.</i></p> <p>Основные идеи функционального подхода. Преимущества и недостатки. Понятие функции. Исчисление лямбда-конверсий. Эволюция языков функционального программирования.</p>	2				<p>Осн. лит.: [1]. Доп. лит.: [1], [3].</p> <p>Библ. online: [3]. Эл. рес.: [3].</p>	
18	<p>Лабораторная работа №9 <i>Структурированный тип данных. Одномерные массивы.</i></p> <p>Понятие массива в языке программирования Pascal. Объявление одномерного массива в программе. Ввод элементов одномерного массива с клавиатуры и вывод на экран. Задание элементов одномерного массива с помощью функции случайных чисел. Процедуры ввода и вывода элементов одномерного массива.</p>		2			Методические указания	Защита отчета по лабораторной работе № 9
19	<p>Лекция № 10 <i>Тема 1.8 Стили и методы программирования.</i></p> <p>Традиционные языки программирования. Сравнение программ на разных языках. Работа программной системы над текстом программы. Модели вычислений и их различия. Особенности элементов традиционной архитектуры вычислительной системы.</p>	2				<p>Осн. лит.: [1]. Доп. лит.: [1], [3].</p> <p>Библ. online: [3]. Эл. рес.: [3].</p>	*Контрольное тестирование №1

1	2	3	4	5	6	7	8
20	Лабораторная работа №10 <i>Двумерные массивы.</i> Математическое понятие двумерного массива. Объявление двумерного массива. Ввод элементов двумерного массива. Вывод элементов двумерного массива на экран. Свойства диагональных элементов.		2			Методические указания	Защита отчета по лабораторной работе № 10
21	Лекция № 11 Модификации традиционной архитектуры. Стековая архитектура. RISC архитектура. Нетрадиционные архитектуры. Многопроцессорность.	2				Осн. лит.: [1]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [3].	*Контрольная работа №1
22	Лабораторная работа №11 <i>Комбинированный тип данных: записи.</i> Понятие записи и поля записи. Объявление типа и переменных этого типа. Массивы записей. Оператор над записями.		2			Методические указания	Защита отчета по лабораторной работе № 6
	Раздел 2. Основы алгоритмизации и программирования	24	24				
23	Лекция № 12 <i>Тема 2.1 Алгоритмизация вычислительных процессов.</i> Основные определения и понятия. Этапы разработки программ. Методы проектирования алгоритмов и программ. Принципы структурного программирования. Понятия алгоритма и его свойства. Способы отображения алгоритмов. Базовые алгоритмические структуры.	2				Осн. лит.: [1]. Доп. лит.: [1], [3], [4]. Библ. online: [3], [4]. Эл. рес.: [1], [3].	Блиц-опрос
24	Лекция № 13 <i>Тема 2.2 Основные элементы языков Pascal и C</i> Общая характеристика языка Pascal. Алфавит языка Pascal. Структура программы. Основные объекты программы. Типы данных и операции, производимые с ними. Стандартные процедуры и функции. Арифметические выражения. Оператор присваивания. Процедуры ввода-вывода данных. Метки и оператор безусловного перехода.	2				Осн. лит.: [1]. Доп. лит.: [1], [3], [4]. Библ. online: [1], [2], [3], [4]. Эл. рес.: [1], [3].	Блиц-опрос

1	2	3	4	5	6	7	8
25	<p>Лабораторная работа №12 <i>Тип данных: файлы данных.</i> Понятие физического и логического файла. Виды файлов данных. Понятие текстового файла данных. Объявление текстового файла. Создание текстового файла данных. Дополнение текстового файла новыми данными. Использование данных из файла. Объявление типизированного файла. Создание типизированного файла. Дополнение типизированного файла новыми данными. Использование данных из типизированного файла.</p>		4			Методические указания	Защита отчета по лабораторной работе № 12
26	<p>Лекция № 14 <i>Тема 2.3 Ветвления. Массивы. Циклы</i> Простые и сложные условия. Операторы ветвления: if и case. Массивы: описание и использование. Операторы цикла: for, while и repeat. Прерывание циклов: break, continue и goto. Решение некоторых типовых задач.</p>	2				Осн. лит.: [1]. Доп. лит.: [1], [3], [4]. Библ. online: [1], [2], [3], [4]. Эл. рес.: [1], [3].	Блиц-опрос
27	<p>Лекция № 15 <i>Тема 2.4 Сортировка массивов</i> Постановка задачи сортировки. Простые алгоритмы сортировки: метод «пузырька», метод «пузырька» с флагом, метод выбора, сортировка простыми вставками, метод прямых вставок с барьером, сортировка бинарными вставками.</p>	2				Осн. лит.: [1]. Доп. лит.: [1], [3]. Библ. online: [1], [2], [3], [4]. Эл. рес.: [1], [3].	Блиц-опрос
28	<p>Лабораторная работа №13 <i>Создание консольных приложений.</i> Знакомство с интегрированной средой разработки. Понятие консольного приложения. Создание нового проекта. Простейшая программа. Компиляция программы. Получение объектного кода. Компоновщик (линковщик). Общая структура программы на C++. Основные директивы и заголовочные файлы. Синтаксис операторов Getch и Cout.</p>		4			Методические указания	Защита отчета по лабораторной работе № 13
29	<p>Лекция № 16 Быстрые алгоритмы сортировки: сортировка Шелла, пирамидальная сортировка, «быстрая» сортировка. Алгоритмы поиска: линейный поиск, двоичный поиск, сравнение методов поиска.</p>	2				Осн. лит.: [1]. Доп. лит.: [1],[3],[4]. Библ. online: [1], [2], [3], [4]. Эл. рес.: [1], [3].	Блиц-опрос

1	2	3	4	5	6	7	8
30	<p>Лекция № 17 <i>Тема 2.5 Строки. Множества</i></p> <p>Символьные и строковые константы. Стандартные функции и процедуры обработки строк. Примеры обработки символьных строк. Множества. Операции с множествами. Примеры использования символов, строк и множеств.</p>	2				<p>Осн. лит.: [1]. Доп. лит.: [1], [3].</p> <p>Библ. online: [1], [2], [3], [4]. Эл. рес.: [3].</p>	Блиц-опрос
31	<p>Лабораторная работа №14 <i>Создание простейших программ линейной структуры.</i></p> <p>Алгоритм линейной программы. Назначение заголовочного файла <math.h>. Способы объявления и инициализации переменных. Синтаксис оператора ввода. Использование арифметических операций и математических функций в выражениях. Использование операций приведения типа. Создание простейших функций в программе. Назначение и использование манипулятора вывода Setw.</p>		4			<p>Методические указания</p>	Защита отчета по лабораторной работе № 14
32	<p>Лекция № 18 <i>Тема 2.6 Подпрограммы в языке Pascal и C</i></p> <p>Структура, назначение и применение подпрограмм. Параметры и аргументы, области действия имен. Процедуры. Функции. Рекурсивные процедуры и функции. Примеры решения задач.</p>	2				<p>Осн. лит.: [1]. Доп. лит.: [1], [3].</p> <p>Библ. online: [1], [2], [3], [4]. Эл. рес.: [1], [3].</p>	Блиц-опрос
33	<p>Лекция № 19 <i>Тема 2.7 Записи</i></p> <p>Понятие записи. Объявление записи и обращение к полям записи. Оператор присоединения With. Массив записей. Записи в процедурах и функциях. Файлы записей. Чтение из файла. Примеры программ. Выделение памяти под запись. Сортировка массива записей.</p>	2				<p>Осн. лит.: [1]. Доп. лит.: [1], [3].</p> <p>Библ. online: [1], [2], [3], [4]. Эл. рес.: [1], [3].</p>	Блиц-опрос
34	<p>Лабораторная работа №15 <i>Создание программ ветвящейся и циклической структуры.</i></p> <p>Синтаксис условных операторов. Использование оператора If. Формы оператора If. Использование оператора Switch. Синтаксис операторов цикла. Использование оператора For. Использование операторов Do и Do-while. Совместное использование ветвящихся и циклических конструкций. Назначение и использование манипуляторов вывода Fixed и Setprecision.</p>		4			<p>Методические указания</p>	Защита отчета по лабораторной работе № 15

1	2	3	4	5	6	7	8
35	<p>Лекция № 20 <i>Тема 2.8 Файлы</i> Общие сведения о файлах и файловой системе. Файлы языка Pascal. Текстовые файлы. Типизированные файлы. Примеры решения задач обработки файлов.</p>	2				<p>Осн. лит.: [1]. Доп. лит.: [1], [3].</p> <p>Библ. online: [1], [2], [3], [4]. Эл. рес.: [1], [3].</p>	Блиц-опрос
36	<p>Лекция № 21 <i>Тема 2.9 Динамические структуры данных</i> Адресация оперативной памяти MS DOS. Указатели и операции над ними. Управление динамической памятью. Работа с динамическими массивами. Динамические структуры данных: списки, стеки, очереди, деки. Деревья и графы. Примеры решения задач обработки динамических структур данных.</p>	2				<p>Осн. лит.: [1]. Доп. лит.: [1], [3].</p> <p>Библ. online: [1], [2], [3], [4]. Эл. рес.: [1], [3].</p>	Блиц-опрос
37	<p>Лабораторная работа №16 <i>Указатели и ссылки.</i> Создание и использование указателей. Понятие и способы объявления указателей. Способы инициализации указателей. Назначение оператора раскрытия указателя (&) и оператора разыменовывания (*). Способы получения адресов переменных. Правила использования арифметических операторов с указателями. Понятие ссылок. Создание и использование ссылок. Указатели и динамическая память.</p>		4			<p>Методические указания</p>	Защита отчета по лабораторной работе № 16
38	<p>Лекция №22 <i>Тема 2.10 Модульное программирование</i> Модули. Открытые массивы и строки. Нетипизированные параметры. Параметры процедурного типа. Примеры решения задач с использованием модулей.</p>	2				<p>Осн. лит.: [1]. Доп. лит.: [1], [3].</p> <p>Библ. online: [1], [2], [3], [4], [5]. Эл. рес.: [1], [3].</p>	*Контрольное тестирование №2
39	<p>Лекция №23 <i>Тема 2.11 Приемы программирования типовых задач</i> Записи. Типизированные файлы. Текстовые файлы. Списки.</p>	2				<p>Осн. лит.: [1]. Доп. лит.: [1], [3].</p> <p>Библ. online: [1], [2], [3], [4], [5]. Эл. рес.: [3].</p>	*Контрольная работа №2

1	2	3	4	5	6	7	8
40	Лабораторная работа №17 <i>Массивы в C++</i> Обработка одномерных статических массивов. Объявление одномерного массива. Обработка двумерных статических массивов. Объявление двумерного массива. Массивы символов. Массивы строк.		4			Методические указания	Защита отчета по лабораторной работе № 17
	Раздел 3. Язык программирования C++. Структурное программирование	22	22				
41	Лекция №24 <i>Тема 3.1 Базовые сведения о языке C++</i> Алфавит языка. Идентификаторы. Ключевые слова. Комментарии. Переменные. Константы и литералы. Преобразование типов.	2				Осн. лит.: [1], [2], [3], [4]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [1], [2].	Блиц-опрос
42	Лекция №25 <i>Тема 3.2 Структура программы на C++</i> Блок заголовков программы. Блок с прототипами. Главная функция программы. Блок с описаниями функций. Примеры программ.	2				Осн. лит.: [1], [2], [3], [4]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [2].	Блиц-опрос
43	Лабораторная работа №18 <i>Массивы и указатели</i> Доступ к элементам массива посредством указателей. Массивы указателей. Массивы в динамической памяти.		4			Методические указания	Защита отчета по лабораторной работе № 18
44	Лекция № 26 <i>Тема 3.3 Вычисления в C++</i> Понятие операции и выражения в C++. Арифметические операции. Операция присваивания. Операции сравнения. Логические операции. Операция размера.	2				Осн. лит.: [1], [2], [3], [4]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [2].	Блиц-опрос
45	Лекция № 27 <i>Тема 3.4 Управляющие инструкции</i> Условные операторы: оператор if, оператор switch. Операторы организации циклов: оператор for, оператор while, оператор do while.	2				Осн. лит.: [1], [2], [3], [4]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [2].	Блиц-опрос

1	2	3	4	5	6	7	8
46	<p>Лабораторная работа №19 <i>Составные типы данных</i> Понятие структуры. Объявление структуры. Присваивание значений полям структурных переменных. Вывод значение полей структурных переменных. Массивы структур. Действия над структурными переменными. Создание и использование объединений. Перечисления.</p>		4			<p>Методические указания</p>	<p>Защита отчета по лабораторной работе № 19</p>
47	<p>Лекция № 28 <i>Тема 3.5 Указатели</i> Понятие переменных-указателей. Операторы, используемые при работе с указателями. Арифметические действия с указателями. Стек и динамически распределяемая память. Указатели и объекты в динамической памяти.</p>	2				<p>Осн. лит.: [1], [2], [3], [4]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [2].</p>	<p>Блиц-опрос</p>
48	<p>Лекция № 29 <i>Тема 3.6 Массивы</i> Одномерные массивы. Двумерные массивы. Инициализация одномерных массивов. Инициализация двумерных массивов. Массивы и указатели. Массивы, расположенные в динамической памяти. Массивы символов. Массив строк.</p>	2				<p>Осн. лит.: [1], [2], [3], [4]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [2].</p>	<p>Блиц-опрос</p>
49	<p>Лабораторная работа №20 <i>Создание и использование функций</i> Понятие функции. Простейшие программы с использованием функций. Прототипы функций. Определение функций. Вызов функций. Функции с параметрами. Создание и использование локальных переменных. Создание и использование глобальных переменных. Перегруженные и рекурсивные функции.</p>		4			<p>Методические указания</p>	<p>Защита отчета по лабораторной работе № 20</p>
50	<p>Лекция № 30 <i>Тема 3.7 Структуры и объединения</i> Определение структуры. Определение структурной переменной. Доступ к полям структуры. Инициализация структурных переменных. Объединения.</p>	2				<p>Осн. лит.: [1], [2], [3], [4]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [2].</p>	<p>Блиц-опрос</p>

1	2	3	4	5	6	7	8
51	Лекция № 31 <i>Тема 3.8 Функции</i> Объявление, определение и вызов функции. Использование аргументов. Использование инструкции return. Версия инструкции return, которая не возвращает значение. Версия инструкции return, которая возвращает значение. Правила действия областей видимости. Локальная область видимости.	2				Осн. лит.: [1], [2], [3], [4]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [2].	Блиц-опрос
52	Лабораторная работа №21 <i>Механизмы передачи аргументов в функции</i> Передача аргумента по значению. Передача аргумента по ссылке. Передача аргументов в функцию по указателю. Способы передачи в функцию массива. Использование безразмерного массива. Передача функциям строк. Возвращение функциями указателей.		4			Методические указания	Защита отчета по лабораторной работе № 21
53	Лекция № 32 Соккрытие имен переменных. Параметры функции. Глобальная область видимости. Перегрузка функций. Способы передачи аргументов в функции. Передача аргумента по значению. Передача аргумента по ссылке. Ссылочные параметры.	2				Осн. лит.: [1], [2], [3], [4]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [2].	*Контрольное тестирование №3
26	Лекция № 33 Передача функции массива. Передача функциям строк. Возвращение функциями указателей и ссылок.	2				Осн. лит.: [1], [2], [3], [4]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [2].	*Контрольная работа №3
27	Лекция № 34 <i>Тема 3.9 Функции и составные типы данных</i> Функции и структуры. Указатель на функцию. Указатели на тип void. Указатели на функции, аргументами или возвращаемыми значениями которых является указатель на void.	2				Осн. лит.: [1], [2], [3], [4]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [2].	
28	Лабораторная работа №22 <i>Выполнение контрольного задания.</i>		6			Методические указания	Защита отчета по лабораторной работе № 22
	Всего (136 часов)	68	68				

* КОНТРОЛЬНЫЕ ТОЧКИ

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

ЛИТЕРАТУРА

Основная:

1. Волкова, Т. И. Введение в программирование [Электронный ресурс]: учебное пособие / Т. И. Волкова. – Москва; Берлин: Директ-Медиа, 2018. – 139 с.: ил., схем., табл. // ЭБС «Университетская библиотека онлайн» – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=493677>.
2. Карпенко, С.Н. Основы объектно-ориентированного программирования на языке С++ [Электронный ресурс]: учебно-методическое пособие. / Карпенко, С.Н. – Нижний Новгород: Нижегородский госуниверситет, 2018. – 104с. // ЭБС «Лань» – Режим доступа: по подписке: URL: <https://e.lanbook.com/book/144808>.
3. Павловская, Т.А. С/С++. Программирование на языке высокого уровня: учебник. – СПб. : Питер, 2021. – 464 с. – Допущено М-вом образования РФ в качестве учебника для студ. вузов, обуч. по направлению подготовки дипломир. специалистов «Информатика и вычисл. Техника».
4. Павловская, Т. А. С/С++. Процедурное и объектно-ориентированное программирование: учебник для вузов.. – СПб : Питер, 2019. – 496 с.

Дополнительная:

1. Кувшинов, Д. Р. Компьютерные науки: Основы программирования: учеб. пособие/ Д- Р. Кувшинов; М-во образования и науки Рос. Федерации, Урал. федер. ун-т. Екатеринбург: Изд-во Урал, ун-та. 2015. – 104 с.
2. Объектно-ориентированное программирование на С++ [Электронный ресурс]: учебник / И. В. Баранова, С. Н. Баранов, И. В. Баженова [и др.]. – Красноярск: СФУ, 2019. – 288 с. // ЭБС «Лань». – Режим доступа: по подписке: URL: <https://e.lanbook.com/book/157572>. – Дата доступа: 19.08.2021.
3. Городняя, Л. В. Парадигма программирования [Электронный ресурс]: учебное пособие для вузов / Л. В. Городняя. – 2-е изд., стер. – Санкт-Петербург: Лань, 2021. – 232 с. // ЭБС «Лань». – URL: <https://e.lanbook.com/book/151660>. – Дата доступа: 19.08.2021.
4. Ульянова, Н. Д. Основные принципы алгоритмизации [Электронный ресурс]: учебно-методическое пособие / Н. Д. Ульянова. – Брянск: Брянский ГАУ, 2020. – 56 с. // ЭБС «Лань». – Режим доступа: по подписке: URL: <https://e.lanbook.com/book/172114>. – Дата доступа: 19.08.2021.
5. ГОСТ 19.701-90. Единая система программной документации. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения.
6. ГОСТ 19781-90. Обеспечение систем обработки информации программное. Термины и определения.

Университетская библиотека онлайн:

1. Балдин, К. В. Математическое программирование [Электронный ресурс] : учебник / К. В. Балдин, Н. А. Брызгалов, А. В. Рукосуев ; под общ. ред. К. В. Балдина. – 2-е изд. – Москва : Дашков и К°, 2018. – 218 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=112201>. – Дата доступа: 19.08.2021.
2. Введение в программные системы и их разработку [Электронный ресурс] / С. В. Назаров, С. Н. Белоусова, И. А. Бессонова [и др.]. – 2-е изд., испр. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 650 с. : схем., табл., ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=429819>. – Дата доступа: 19.08.2021.

С. В. Тужикова

ПЕРЕЧЕНЬ ТЕМ ЛАБОРАТОРНЫХ ЗАНЯТИЙ

Лабораторная работа №1. Введение в Pascal.

Знакомство с интерфейсом программы PascalABC.NET. Базовые навыки написания программ на языке Pascal. Ввод и вывод данных с консоли. Простейшие операции.

Лабораторная работа №2. Выражения, их типы.

Алфавит языка. Арифметические выражения. Некоторые стандартные функции, типы аргумента и результаты. Операции отношений и логические операции. Приоритет операций.

Лабораторная работа №3. Программирование алгоритмов линейной структуры.

Окно редактора кода. Окно вывода. Окно ввода. Запуск и остановка программы. Структура программы. Константы и переменные. Целочисленные типы данных. Вещественный тип данных. Символьный тип данных. Логический тип данных. Процедуры ввода-вывода данных. Оператор присваивания. Пустой оператор.

Лабораторная работа №4. Программирование алгоритмов разветвляющейся структуры.

Оператор условного перехода If. Оператор выбора Case. Составной оператор.

Лабораторная работа №5. Программирование алгоритмов циклической структуры.

Понятие тела цикла. Оператор цикла While. Оператор цикла Repeat. Оператор цикла For. Способы отображения на блок-схемах.

Лабораторная работа №6. Программирование алгоритмов с использованием вложенных циклов.

Понятие вложенного цикла. Внешний и внутренний циклы. Правила организации вложенных циклов. Особенности отображения на блок-схемах.

Лабораторная работа №7. Строковые переменные.

Данные строкового типа. Объявление строковой переменной. Строковые выражения. Операции присваивания, объединения, сравнения и отношения над строками. Стандартные функции для работы со строками. Стандартные процедуры для работы со строками.

Лабораторная работа №8. Программирование с использованием пользовательских процедур и функций.

Понятие подпрограммы. Понятие процедуры. Понятие функции. Встроенные и определённые пользователем процедуры и функции.

Лабораторная работа №9. Структурированный тип данных. Одномерные массивы.

Понятие массива в языке программирования Pascal. Объявление одномерного массива в программе. Ввод элементов одномерного массива с клавиатуры и вывод на экран. Задание элементов одномерного массива с помощью функции случайных чисел. Процедуры ввода и вывода элементов одномерного массива.

Лабораторная работа №10. Двумерные массивы.

Математическое понятие двумерного массива. Объявление двумерного массива. Ввод элементов двумерного массива. Вывод элементов двумерного массива на экран. Свойства диагональных элементов.

Лабораторная работа №11. Комбинированный тип данных: записи.

Понятие записи и поля записи. Объявление типа и переменных этого типа. Массивы записей. Оператор над записями.

Лабораторная работа №12. Тип данных: файлы данных.

Понятие физического и логического файла. Виды файлов данных. Понятие текстового файла данных. Объявление текстового файла. Создание текстового файла данных. Дополнение текстового файла новыми данными. Использование данных из файла. Объявление типизированного файла. Создание типизированного файла. Дополнение типизированного файла новыми данными. Использование данных из типизированного файла.

Лабораторная работа №13. Создание консольных приложений.

Знакомство с интегрированной средой разработки. Понятие консольного приложения. Создание нового проекта. Простейшая программа. Компиляция программы. Получение объектного кода. Компоновщик (линковщик). Общая структура программы на C++. Основные директивы и заголовочные файлы. Синтаксис операторов Getch и Cout.

Лабораторная работа №14. Создание простейших программ линейной структуры.

Алгоритм линейной программы. Назначение заголовочного файла <math.h>. Способы объявления и инициализации переменных. Синтаксис оператора ввода. Использование арифметических операций и математических функций в выражениях. Использование операций приведения типа. Создание простейших функций в программе. Назначение и использование манипулятора вывода Setw.

Лабораторная работа №15. Создание программ ветвящейся и циклической структуры.

Синтаксис условных операторов. Использование оператора If. Формы оператора If. Использование оператора Switch. Синтаксис операторов цикла. Использование оператора For. Использование операторов Do и Do-while. Совместное использование ветвящихся и циклических конструкций. Назначение и использование манипуляторов вывода Fixed и Setprecision.

Лабораторная работа №16. Указатели и ссылки.

Создание и использование указателей. Понятие и способы объявления указателей. Способы инициализации указателей. Назначение оператора раскрытия указателя (&) и оператора разыменовывания (*). Способы получения адресов переменных. Правила использования арифметических операторов с указателями. Понятие ссылок. Создание и использование ссылок. Указатели и динамическая память.

Лабораторная работа №17. Массивы в C++

Обработка одномерных статических массивов. Объявление одномерного массива. Обработка двумерных статических массивов. Объявление двумерного массива. Массивы символов. Массивы строк.

Лабораторная работа №18. Массивы и указатели

Доступ к элементам массива посредством указателей. Массивы указателей. Массивы в динамической памяти.

Лабораторная работа №19. Составные типы данных

Понятие структуры. Объявление структуры. Присваивание значений полям структурных переменных. Вывод значение полей структурных переменных. Массивы структур. Действия над структурными переменными. Создание и использование объединений. Перечисления.

Лабораторная работа №20. Создание и использование функций

Понятие функции. Простейшие программы с использованием функций. Прототипы функций. Определение функций. Вызов функций. Функции с параметрами. Создание и использование локальных переменных. Создание и использование глобальных переменных. Перегруженные и рекурсивные функции.

Лабораторная работа №21. Механизмы передачи аргументов в функции

Передача аргумента по значению. Передача аргумента по ссылке. Передача аргументов в функцию по указателю. Способы передачи в функцию массива. Использование безразмерного массива. Передача функциям строк. Возвращение функциями указателей.

Лабораторная работа №22. Выполнение контрольного задания.

ПЕРЕЧЕНЬ ВОПРОСОВ ДЛЯ ПРОВЕДЕНИЯ ЗАЧЕТА

1. Формирование стратегии ввода исходных данных и вывода результатов.
2. Проверка сложных условий.
3. Обработка цифр целых и вещественных чисел любой разрядности.
4. Построение таблиц значений функций.
5. Формирование числовых последовательностей по рекуррентной формуле или с заданным свойством, расчет сумм, произведений, среднего значения и пр.
6. Вычисление суммы (произведения) указанного количества членов ряда.
7. Вычисление суммы (произведения) членов ряда с заданной точностью.
8. Поиск корней уравнений с заданной точностью.
9. Обработка одномерных массивов (перебор всех элементов массива или некоторой части; поиск элементов с заданным свойством; преобразование, удаление, вставка и перестановка элементов; сортировка всего массива или некоторой его части; подсчет характеристических свойств и т.п.).
10. Обработка матрицы (перебор всей или части матрицы: относительно ее диагоналей, части строк, столбцов или элементов; поиск элементов с заданным свойством; преобразование, перестановка, удаление и вставка элементов, строк, столбцов; сортировка всей матрицы или некоторой части; подсчет характеристических свойств и т.п.).
11. Обработка нескольких массивов одновременно (слияние, перемещение, поиск и т.п.).
12. Обработка символов и строк (обработка текста как линейного массива; обработка с помощью стандартных процедур и функций; поиск символов или слов с заданным свойством; их преобразование, перестановка, удаление и вставка; сортировка символов и слов; подсчет характеристических свойств и т.п.).

ПЕРЕЧЕНЬ ВОПРОСОВ ДЛЯ ПРОВЕДЕНИЯ ЭКЗАМЕНА

13. Этапы разработки программ.
14. Методы проектирования программ.
15. Система программирования, характеристика основных элементов.
16. Классификация средств представления алгоритмов.
17. Основные элементы блок-схем.
18. Классификация базовых канонических структур алгоритмов.
19. Характеристика линейных и разветвляющихся алгоритмов.
20. Характеристика циклических алгоритмов
21. Структура программы на языках Pascal и C.
22. Классификация типов данных языках Pascal и C.
23. Совместимость и эквивалентность типов данных языка Pascal и C.
24. Простые и сложные условия языка Pascal и C.
25. Метод дихотомии в задаче поиска корней уравнения.
26. Способы перебора элементов массива.
27. Постановка задачи сортировки. Классификация алгоритмов сортировки.
28. Метод «пузырька» и его модификации.
29. Сортировка выбором.
30. Сортировка простыми вставками.
31. Сортировка простыми вставками с барьером.
32. Сортировка бинарными вставками.
33. Сортировка Шелла.
34. Пирамидальная сортировка.
35. Быстрая сортировка.
36. Двоичный (бинарный) поиск.
37. Объявление функции.
38. Объявления процедуры.
39. Способы подстановки аргументов в подпрограммах.
40. Области действия имен в программах.
41. Рекурсивные процедуры и функции.
42. Символьные и строковые константы.
43. Стандартные функции и процедуры обработки строк.
44. Множества.
45. Записи.
46. Общие сведения о файлах и файловой системе.
47. Файлы языка Pascal.
48. Текстовые файлы.
49. Типизированные файлы.
50. Адресация оперативной памяти MS DOS.
51. Указатели и операции над ними.
52. Управление динамической памятью.
53. Работа с динамическими массивами.
54. Динамические структуры данных: списки, стеки, очереди, деки.
55. Деревья и графы.
56. Модули.
57. Открытые массивы и строки.
58. Нетипизированные параметры.
59. Параметры процедурного типа.
60. Последовательность создания проекта, компиляция и запуск приложения в Visual C++. Общая структура программы. Назначение директивы `#include` и заголовочных файлов `<iostream>` и `<conio.h>`.

61. Алфавит языка C++. Идентификаторы. Ключевые слова. Комментарии.
62. Переменные языка C++. Константы и литералы. Преобразования типов в языке C++.
63. Структура программы на языке C++.
64. Вычисления в C++. Арифметические операции. Операции присваивания. Операции сравнения. Операции размера.
65. Условные операторы в C++.
66. Операторы организации циклов в C++.
67. Понятие переменных-указателей в C++, операторы, используемые при работе с указателями.
68. Арифметические действия с указателями в C++.
69. Стек и динамически распределяемая память в C++.
70. Указатели и объекты в динамической памяти в C++.
71. Одномерные массивы в C++. Инициализация.
72. Двумерные массивы в C++. Инициализация.
73. Массивы и указатели в C++.
74. Массивы в динамической памяти в C++.
75. Массивы символов в C++.
76. Массивы строк в C++.
77. Определение структуры, структурной переменной, доступ к полям структуры в C++.
78. Инициализация структурных переменных. Объединения в C++.
79. Объявление, определение и вызов функции в C++.
80. Использование инструкции return.
81. Области видимости переменных, параметры функции в C++.
82. Перегрузка функций в C++.
83. Способы передачи аргументов в функцию в C++.
84. Передача в функцию в C++ массива.
85. Передача в функцию в C++ строк.
86. Возвращение функциями в C++ указателей и ссылок.
87. Указатели на функцию в C++.

ОРГАНИЗАЦИЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Обучение дисциплине «Основы и методологии программирования» предполагает реализацию следующих форм самостоятельной работы студентов:

- проработка конспекта лекций и учебной литературы;
- изучение печатных источников по теме дисциплины;
- изучение профессиональных электронных ресурсов по теме дисциплины;
- подготовка к аудиторному выполнению лабораторных работ (предварительное знакомство с методическими указаниями, программным обеспечением, вариантом индивидуального задания по работе);
- решение индивидуальных задач при подготовке к лабораторным занятиям;
- выполнение практических упражнений (работа с тренажерами) для закрепления знаний и навыков;
- подготовка к защите лабораторных работ (оформление отчёта по индивидуальному варианту задания, защита результатов работы и демонстрации степени освоения навыков и умений по конкретной теме);
- решение во внеурочное время контрольных задач, получаемых на лекциях;
- углублённое изучение отдельных тем учебной дисциплины для подготовки к устным опросам;
- изучение основной и дополнительной литературы в процессе подготовки к анализу и решению проблемных задач, реализации элементов исследовательской деятельности;
- проведение самостоятельных исследований по конкретным тематическим направлениям;
- подготовка к промежуточной и текущей диагностике компетенции;
- систематизация полученных знаний при подготовке к зачету и экзамену.

Условия для самостоятельной работы студентов, в частности, для развития навыков самоконтроля, способствующих интенсификации образовательного процесса, обеспечиваются:

- использованием бизнес-мессенджера для групповой работы и общения Microsoft Teams;
- наличием и полной доступностью электронных вариантов курса лекций и учебно-методического пособия по основным разделам дисциплины.

Дополнительное учебно-методическое обеспечение самостоятельной работы студентов очной формы обучения

Материалы, размещённые в бизнес-мессенджере для групповой работы и общения Microsoft Teams: шифр курса А6ТАМВФ.

Методические указания к выполнению лабораторных работ по дисциплине «Основы и методологии программирования» для студентов специальности 1-98 01 01 «Компьютерная безопасность (по направлениям)».

Содержание самостоятельной работы студентов

Вид самостоятельной работы	Тематическое содержание и используемые источники	Количество часов
1	2	3
Самостоятельное изучение отдельных вопросов по темам дисциплины при подготовке к контрольным работам	<p><i>Тема 2.1 Алгоритмизация вычислительных процессов.</i> Принципы структурного программирования. Осн. лит.: [1]. Доп. лит.: [1], [3], [4]. Библ. online: [3], [4]. Эл. рес.: [1], [3].</p>	2
	<p><i>Тема 2.3 Ветвления. Массивы. Циклы</i> Решение некоторых типовых задач. Осн. лит.: [1]. Доп. лит.: [1], [3], [4]. Библ. online: [1], [2], [3], [4]. Эл. рес.: [1], [3].</p>	2
	<p><i>Тема 2.4 Сортировка массивов</i> Простые алгоритмы сортировки. Быстрые алгоритмы сортировки. Осн. лит.: [1]. Доп. лит.: [1], [3]. Библ. online: [1], [2], [3], [4]. Эл. рес.: [1], [3].</p>	2
	<p><i>Тема 2.9 Динамические структуры данных</i> Примеры решения задач обработки динамических структур данных. Осн. лит.: [1]. Доп. лит.: [1], [3]. Библ. online: [1], [2], [3], [4]. Эл. рес.: [1], [3].</p>	2
	<p><i>Тема 2.10 Модульное программирование</i> Примеры решения задач с использованием модулей. Осн. лит.: [1]. Доп. лит.: [1], [3]. Библ. online: [1], [2], [3], [4], [5]. Эл. рес.: [1], [3].</p>	2
	<p><i>Тема 3.1 Базовые сведения о языке C++</i> Примеры программ. Осн. лит.: [1], [2], [3], [4]. Доп. лит.: [1], [3]. Библ. online: [3]. Эл. рес.: [1], [2].</p>	2
Подготовка к защите отчетов по лабораторным работам	<p><i>Лабораторная работа №1 Введение в Pascal.</i></p>	1
	<p><i>Лабораторная работа №2 Выражения, их типы.</i></p>	1
	<p><i>Лабораторная работа №3 Программирование алгоритмов линейной структуры.</i></p>	1
	<p><i>Лабораторная работа №4 Программирование алгоритмов разветвляющейся структуры.</i></p>	1
	<p><i>Лабораторная работа №5 Программирование алгоритмов циклической структуры.</i></p>	1
	<p><i>Лабораторная работа №6 Программирование алгоритмов с использованием вложенных циклов.</i></p>	1
	<p><i>Лабораторная работа №7 Строковые переменные.</i></p>	1
	<p><i>Лабораторная работа №8. Программирование с использованием пользовательских процедур и функций.</i></p>	1
	<p><i>Лабораторная работа №9 Структурированный тип данных. Одномерные массивы.</i></p>	1
	<p><i>Лабораторная работа №10 Двумерные массивы.</i></p>	1
	<p><i>Лабораторная работа №11 Комбинированный тип данных: записи.</i></p>	1
	<p><i>Лабораторная работа №12 Тип данных: файлы данных.</i></p>	1
	<p><i>Лабораторная работа №13 Создание консольных приложений.</i></p>	2
	<p><i>Лабораторная работа №14 Создание простейших программ линейной структуры.</i></p>	2
	<p><i>Лабораторная работа №15 Создание программ ветвящейся и циклической структуры.</i></p>	2

1	2	3
Подготовка к защите отчетов по лабораторным работам	<i>Лабораторная работа №16 Указатели и ссылки.</i>	2
	<i>Лабораторная работа №17 Массивы в C++</i>	2
	<i>Лабораторная работа №18 Массивы и указатели</i>	2
	<i>Лабораторная работа №19 Составные типы данных</i>	2
	<i>Лабораторная работа №20 Создание и использование функций</i>	2
	<i>Лабораторная работа №21 Механизмы передачи аргументов в функции</i>	2
	<i>Лабораторная работа №22 Выполнение контрольного задания</i>	2
Систематизация полученных знаний при подготовке к экзамену		36
		ВСЕГО 80

КОНТРОЛЬ КАЧЕСТВА УСВОЕНИЯ ЗНАНИЙ

Учебном плане специальности в качестве формы текущей аттестации по учебной дисциплине «Основы и методологии программирования» предусмотрены две формы контроля зачет и экзамен.

Оценка учебных достижений студента производится по десятибалльной шкале.

Диагностика качества усвоения знаний проводится в соответствии с Положением о рейтинговой системе оценки знаний и компетенций студентов (приказ ректора УО ПГУ № 294 от 06.06.2014 (в редакции, утверждённой приказом № 605 от 17.11.2014) в форме промежуточного контроля и текущей аттестации.

Для оценивания самостоятельной и аудиторной работы студентов в рамках курса для контроля успеваемости используется накопительная система, которая предполагает суммирование отметок, выставляемых в электронный журнал за все виды работ в течение прохождения для определения среднеарифметических показателей успеваемости.

Мероприятия промежуточного контроля проводятся в течение семестра и включают в себя следующие формы контроля:

- устная форма (блиц-опрос на лекциях);
- письменная форма (тесты, контрольные работы);
- устно-письменная форма (отчёты по лабораторным с их устной защитой);
- техническая форма (электронные тесты, визуальные лабораторные работы представленные в виде кода на языке программирования, тестовая форма проведения экзамена).

Лабораторные работы предполагают выполнение и защиту. Последнее занятие каждой из активностей в семестре предусматривает выполнение и защиту зачётной итоговой работы. При выполнении лабораторных работ выдаётся индивидуальное задание. Отчёты по лабораторным работам представляются в электронном виде. Содержание отчёта: название работы, вариант задания, анализ задания, ход выполнения работы, основные и промежуточные результаты, выводы по работе. Защита работ проводится индивидуально и оценивается в соответствии установленными правилами.

Промежуточная (аттестационная) диагностика компетенции студентов осуществляется на основании индивидуального рейтинга студента на момент аттестации. Для положительной аттестации (промежуточного контроля успеваемости) необходимо согласно календарному плану выполнить все лабораторные работы (продумать задания, описанные в методических указаниях и выполнить индивидуальные задания), а также иметь положительную оценку по промежуточному контролю освоения теоретической части курса.

Результат промежуточного контроля за семестр оценивается отметкой в баллах по десятибалльной шкале и выводится, исходя из отметок, выставленных в ходе проведения мероприятий промежуточного контроля в течение семестра по следующей формуле:

$$\Pi = \frac{(KT_1 + \dots + KT_n) + (LP_1 + \dots + LP_{22}) + (KP_1 + KP_2 + KP_3)}{(25 + n)},$$

где $KT_1 + \dots + KT_n$ – отметки, выставленные по результатам контрольного тестирования;
 n – количество тестов (три теста являются обязательными, остальные представлены в виде тренингов);

$LP_1 + \dots + LP_{22}$ – отметки, выставленные по результатам защит лабораторных работ.

KP_1, KP_2, KP_3 – отметки, выставленные по результатам контрольных работ.

Результат промежуточного контроля рассчитывается как округлённое среднее значение. Результат может быть увеличен в соответствии с п.п. 6.8 и 6.9 Положения.

Текущая аттестация проводится в форме зачёта и экзамена.

Зачёт проводится согласно Положению.

Заключение о зачёте формируется на основе накопительного принципа по формуле:

$$З = k \cdot П,$$

где k – весовой коэффициент промежуточного контроля;

$П$ – результат промежуточного контроля за семестр.

Весовой коэффициент k принимается равным 1.

Если полученная отметка $З < 4$ баллов, то проводится устный зачёт отдельно по представленным в программе вопросам.

Экзамен проводится согласно Положению.

Итоговая экзаменационная отметка (ИЭ) учитывает отметку по результатам промежуточного контроля ($П$) и экзаменационную отметку ($Э$). Весовой коэффициент k принимается равным 0,5. Информация о весовом коэффициенте доводится до студентов на первом занятии в семестре. Составляющие для формирования итоговой отметки по дисциплине и их весовые коэффициенты представлены в таблице 1.

Таблица 1 – Составляющие итоговой отметки по дисциплине

Составляющие (ИЭ)	k	$П$	$1-k$	$Э$
	0,5	Представлены в таблице 2	0,5	*

* Отметка, полученная студентом на экзамене за письменный ответ по экзаменационному билету.

В таблице 1 представлены составляющие, формирующие отметку промежуточного контроля $П$ по дисциплине.

Таблица 1 – Составляющие отметки промежуточного контроля $П$ по дисциплине

Промежуточные контрольные мероприятия	Содержание контрольного мероприятия – название раздела (темы)	Задания контрольного мероприятия	Отметка контрольных мероприятий (КР), (КТ), (ЛР)
Контрольная работа №1	Тема 1.8 Стили и методы программирования.	Предлагается ответить на вопросы.	Максимальная оценка 10 (десять) баллов
Контрольная работа №2	Тема 2.11 Приемы программирования типовых задач	Предлагается ответить на вопросы.	Максимальная оценка 10 (десять) баллов
Контрольная работа №3	Тема 3.8 Функции Механизмы передачи аргументов в функции	Предлагается ответить на вопросы.	Максимальная оценка 10 (десять) баллов
Контрольный тест	Темы и планируемые контрольные тесты указаны в учебно-методической карте дисциплины	Тест ориентирован на прохождение в online-режиме и оформлен в Forms и размещен бизнес-мессенджере для групповой работы и общения Microsoft Teams	Максимальная оценка 10 (десять) баллов

Итоговая отметка по дисциплине определяется по формуле:

$$ИЭ = 0,5П + 0,5Э.$$

Положительной является экзаменационная отметка не ниже 4 баллов.

ХАРАКТЕРИСТИКА ИННОВАЦИОННЫХ ПОДХОДОВ К ПРЕПОДАВАНИЮ УЧЕБНОЙ ДИСЦИПЛИНЫ

Основные методы (технологии) обучения, отвечающие целям и задачам учебной дисциплины:

- проблемное обучение (проблемное изложение, вариативное изложение), реализуемое на лекционных занятиях;
- учебно-исследовательская деятельность, реализация творческого подхода, реализуемые на лабораторных занятиях;

Используемые технологии обучения и диагностики компетенций в преподавании дисциплины «Основы и методологии программирования» реализуют подход, основанный на максимально возможном использовании внутренней и учебной мотивации студента, проявляющейся в чётком понимании им значимости всех видов выполняемых работ, как с точки зрения важности для профессиональной подготовки, так и с точки зрения оценивания. Подход предполагает использование элементов проблемного обучения и элементов исследовательской деятельности студентов в процессе аудиторной работы, а также при выполнении самостоятельных работ при постоянном рейтинговом контроле.

На лекционных занятиях по дисциплине «Основы и методологии программирования» возможно использование элементов проблемного обучения: проблемное изложение некоторых аспектов, использование частично-поискового метода.

Изучение учебной дисциплины осуществляется на лекционных и лабораторных занятиях. На лекционных занятиях студенты овладевают системой теоретических знаний в области решения научно-технических и информационных задач с использованием высокоуровневых алгоритмических языков программирования. В ходе лекционного изложения теоретических сведений используются традиционные словесные приёмы и методы, которые активизируются постановкой проблемных вопросов и заданий, организацией учебных дискуссий с опорой на имеющуюся начальную подготовку студентов и их политехнический кругозор, использованием интерактивных методов обучения.

На лабораторных занятиях развиваются и формируются необходимые практические умения и навыки программирования, а также формируются навыки решения задач в различных прикладных областях.

Также во время проведения лабораторных работ особое внимание уделяется формированию у студентов умения планировать свою работу и определять эффективную последовательность её выполнения.

**ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ
С ДРУГИМИ УЧЕБНЫМИ ДИСЦИПЛИНАМИ СПЕЦИАЛЬНОСТИ**

Название дисциплины, по которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы по изучаемой учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу
«Разработка кросс- платформенных приложений»	Математики и компьютерной безопасности	<i>нет</i>	
«Машинно- ориентированное программирование»	Математики и компьютерной безопасности	<i>нет</i>	
«Технологии программирования»	Математики и компьютерной безопасности	<i>нет</i>	
«Алгоритмы и структуры данных»	Математики и компьютерной безопасности	<i>нет</i>	
«Базы данных»	Математики и компьютерной безопасности	<i>нет</i>	
«Модели данных и системы управления базами данных»	Математики и компьютерной безопасности	<i>нет</i>	

Заведующий кафедрой математики и
компьютерной безопасности, к.ф.-м.н., доцент



И.Б. Бураченко

**ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ
С ДРУГИМИ УЧЕБНЫМИ ДИСЦИПЛИНАМИ СПЕЦИАЛЬНОСТИ**

Название дисциплины, по которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы по изучаемой учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу
«Промышленное программирование»	Энергетики и электроники	<i>Предложений и замечаний нет</i>	

Заведующий кафедрой энергетики и
электроники, к.ф.-м.н., доцент



Д.А. Довгяло

РЕЦЕНЗИЯ

на учебную программу учреждения высшего образования
по учебной дисциплине «Основы и методологии программирования»
для специальности 1-98 01 01 Компьютерная безопасность (по направлениям)
направление специальности 1-98 01 01-01 Компьютерная безопасность
(математические методы и программные системы)
специализация 1-98 01 01-01 03 Защищенные информационные системы»,
подготовленную к.т.н., доцентом Бураченко И.Б.

Разработанная программа составлена на основе учебного плана специальности 1-98 01 01 дневной очной формы получения высшего образования и полностью соответствует современным требованиям, предъявляемым к высшей школе. В программе достаточно четко и подробно сформулированы цели и задачи дисциплины, требования к знаниям студентов и практические навыки, которые необходимо дать студентам при изучении данной дисциплины.

Разработанная программа реализует подход, основанный на максимально возможном использовании внутренней и учебной мотивации студента, проявляющейся в четком понимании им значимости всех видов выполняемых работ, как с точки зрения важности для профессиональной подготовки.

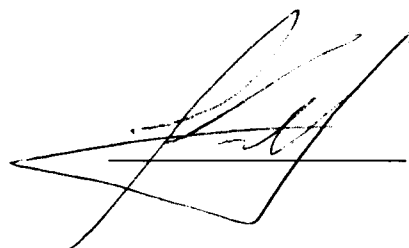
Программой предусмотрено закрепление знаний, полученных на лекционных занятиях с использованием проблемного обучения.

Успешность реализации дисциплины обеспечивается за счет применения на учебных занятиях разнообразных методов организации учебного процесса и современных педагогических технологий обучения с целью развития у студентов математического, логико-алгоритмического и программистского стилей мышления и активного привлечения студентов к учебно-исследовательской деятельности. Достаточная глубина содержания дисциплины по всем разделам и темам учебной программы, соответствие минимуму содержания, который установлен обязательным стандартом по специальности 1-98 01 01.

Дисциплина разбита на 3 модуля, равноценных по объему и логически связанных по содержанию. Темы лекций достаточно подробно расшифрованы с полным отображением их содержания. Лабораторные занятия увязаны с лекционным материалом и вытекают из его содержания. Большая часть лабораторных занятий связана с изучением научно-технических и информационных задач и их решением с использованием высокоуровневых алгоритмических языков программирования. Разработанный план самостоятельной работы дополняет аудиторные занятия и его выполнение позволит студентам более глубоко изучить вопросы методологии программирования.

Считаю, программу учреждения высшего образования по учебной дисциплине «Основы и методологии программирования» соответствующей уровню подготовки студентов в вузах и рекомендую ее к использованию в учебном процессе.

Lead Software at .Net Division
ООО «Godel Technologies Europe»



В.О. Козлов

РЕЦЕНЗИЯ

на учебную программу учреждения высшего образования
по учебной дисциплине «Основы и методологии программирования»
для специальности 1-98 01 01 Компьютерная безопасность (по направлениям)
направление специальности 1-98 01 01-01 Компьютерная безопасность
(математические методы и программные системы)
специализация 1-98 01 01-01 03 Защищенные информационные системы»,
подготовленную к.т.н., доцентом Бураченко И.Б.

Разработанная программа составлена на основе учебного плана специальности 1-98 01 01 дневной очной формы получения высшего образования и полностью соответствует современным требованиям, предъявляемым к высшей школе.

В программе достаточно четко и подробно сформулированы цели и задачи дисциплины, требования к знаниям студентов и практические навыки, которые необходимо дать студентам при изучении данной дисциплины.

Разработанная программа реализует подход, основанный на максимально возможном использовании внутренней и учебной мотивации студента, проявляющейся в четком понимании им значимости всех видов выполняемых работ, как с точки зрения важности для профессиональной подготовки.

Программой предусмотрено закрепление знаний, полученных на лекционных занятиях с использованием проблемного обучения. Также предлагается внедрение активных методов обучения в рамках лабораторных работ для решения практических и ситуационных задач с целью развития у студентов математического, логико-алгоритмического и программистского стилей мышления и активного привлечения студентов к учебно-исследовательской деятельности.

Дисциплина разбита на 3 модуля, равноценных по объему и логически связанных по содержанию. Темы лекций достаточно подробно расшифрованы с полным отображением их содержания. Лабораторные занятия увязаны с лекционным материалом и вытекают из его содержания. Большая часть лабораторных занятий связана с изучением научно-технических и информационных задач и их решением с использованием высокоуровневых алгоритмических языков программирования.

Составленный план самостоятельной работы дополняет аудиторные занятия и его выполнение позволит студентам более глубоко изучить вопросы программирования задач в различных областях.

Критерии оценки знаний студентов изложены в общем плане и адаптированы к модульно-рейтинговой системе обучения.

Считаю, программу учреждения высшего образования по учебной дисциплине «Основы и методологии программирования» соответствующей уровню подготовки студентов в вузах и рекомендую ее к использованию в учебном процессе.

Технический директор
ООО «ТриИнком», к.т.н., доцент



К.Я. Раханов