

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Степанов Павел Иванович

Должность: Руководитель НТИ НИЯУ МИФИ

Дата подписания: 23.02.2026 22:13:07

Уникальный программный ключ:

8c65c591e26b2d8e4609277401f101a71315

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Национальный исследовательский ядерный университет «МИФИ»

Новоуральский технологический институт –

филиал федерального государственного автономного образовательного учреждения
высшего образования «Национальный исследовательский ядерный университет
«МИФИ»

УТВЕРЖДЕНА

Ученым советом НТИ НИЯУ МИФИ

Протокол № 3 от 24.04.2023

Рабочая программа учебной дисциплины "Технологии программирования"

Направление подготовки 09.03.01 – Информатика и вычислительная техника

Профиль подготовки Автоматизированные системы обработки информации и управления

Квалификация (степень) выпускника Академический бакалавр

Форма обучения очная

	Очная форма обучения
Семестр	7
Трудоемкость, ЗЕТ	4 ЗЕТ
Трудоемкость, ч.	144 ч.
Аудиторные занятия, в т.ч.:	72 ч.
- лекции	18 ч.
- лабораторные работы	36 ч.
- практические работы	18 ч.
Самостоятельная работа	27 ч.
Контроль	45 ч.
Форма итогового контроля	экзамен

Программу составила
старший преподаватель кафедры АУ



Тихонова Е.В.

СОДЕРЖАНИЕ

1 ЦЕЛИ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ.....	4
2 МЕСТО УЧЕБНОЙ ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВПО	4
3 ФОРМИРУЕМЫЕ КОМПЕТЕНЦИИ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ	4
4 ВОСПИТАТЕЛЬНЫЙ ПОТЕНЦИАЛ ДИСЦИПЛИНЫ	5
5 СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ.....	8
5.1 Структура курса «Технологии программирования»	8
5.2 Содержание лекционных занятий (7-й семестр) – 18 часов	9
5.3 Темы лабораторных занятий (7-й семестр) – 36 часов	10
5.4 Темы практических занятий (7-й семестр) – 18 часов	10
5.5 Самостоятельная работа – 27 часов	11
6 ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ.....	12
7 СРЕДСТВА ДЛЯ КОНТРОЛЯ И ОЦЕНКИ.....	13
8 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ.....	14
9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	15
Приложение 1. Перечень учебно-методического обеспечения самостоятельной работы студентов.....	17
Приложение 2. Методические указания для студентов по освоению дисциплины.....	18
Приложение 3. Балльно-рейтинговая система оценки.....	19
Приложение 4. Фонд оценочных средств	20

1 ЦЕЛИ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Целью освоения учебной дисциплины «Технологии программирования» является изучение теоретических основ и приобретение практических навыков разработки программного обеспечения (ПО), базирующихся на фундаментальных принципах построения программного продукта (ПП) и объектно-ориентированной парадигме программирования (ООП).

2 МЕСТО УЧЕБНОЙ ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВПО

В соответствии с кредитно-модульной системой подготовки бакалавров по направлению подготовки ВПО 09.03.01 «Информатика и вычислительная техника» профиля подготовки бакалавров «Автоматизированные системы обработки информации и управления» учебная дисциплина «Технологии программирования» относится к обязательным дисциплинам вариативной части раздела «Б1.В.01.02».

Дисциплина «Технологии программирования» входит в число дисциплин формирования профессиональных компетенций выпускника и служит опорой для подготовки к его итоговой государственной аттестации.

Дисциплина знакомит с принципами разработки и сопровождения программного обеспечения. Предшествующий уровень образования обучаемого – среднее (полное) общее образование.

3 ФОРМИРУЕМЫЕ КОМПЕТЕНЦИИ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Профессиональные компетенции (ПК) в соответствии с задачами и объектами (областями знаний) профессиональной деятельности:

Код и наименование профессиональной компетенции;	Код и наименование индикатора достижения профессиональной компетенции
ПК-10.1 Способен разрабатывать и тестировать прототип информационной системы в соответствии с требованиями технического задания	З-ПК-10.1 Знать: языки программирования и работы с базами данных, основы современных операционных систем, основы современных систем управления базами данных, современные объектно-ориентированные языки программирования, устройство и функционирование современных ИС. У-ПК-10.1 Уметь: кодировать на языках программирования, тестировать результаты собственной работы. В-ПК-10.1 Владеть: методами разработки кода прототипа ИС и баз данных прототипа в соответствии с трудовым заданием, проведения тестирования.
ПК-10.3 Способен создавать программный код в соответствии с техническим заданием (готовыми спецификациями)	З-ПК-10.3 Знать: основы программирования, современные объектно-ориентированные языки программирования, современные структурные языки программирования. У-ПК-10.3 Уметь: кодировать на языках программирования, тестировать результаты кодирования. В-ПК-10.3 Владеть: методами разработки кода ИС

	и баз данных ИС, верификации кода ИС и баз данных ИС относительно дизайна ИС и структуры баз данных ИС, устранения обнаруженных несоответствий.
--	---

4 ВОСПИТАТЕЛЬНЫЙ ПОТЕНЦИАЛ ДИСЦИПЛИНЫ

Цели и задачи воспитания, воспитательный потенциал дисциплин:

Направления/цели воспитания	Задачи воспитания (код)	Воспитательный потенциал дисциплин
Профессиональное воспитание	В17 Формирование чувства личной ответственности за научно-технологическое развитие России, за результаты исследований и их последствия	1.Использование воспитательного потенциала дисциплин профессионального модуля для формирования чувства личной ответственности за достижение лидерства России в ведущих научно-технических секторах и фундаментальных исследованиях, обеспечивающих ее экономическое развитие и внешнюю безопасность, посредством и практической значимости результатов научных исследований и технологических разработок. 2. Использование воспитательного потенциала дисциплин профессионального модуля для формирования социальной ответственности ученого за результаты исследований и их последствия, развития исследовательских качеств посредством выполнения учебно-исследовательских заданий, ориентированных на изучение и проверку научных фактов, критический анализ публикаций в профессиональной области, вовлечения в реальные междисциплинарные научно-исследовательские проекты.
	В18 Формирование ответственности за профессиональный выбор, профессиональное развитие и профессиональные решения	Использование воспитательного потенциала дисциплин профессионального модуля для формирования у студентов ответственности за свое профессиональное развитие посредством выбора студентами индивидуальных образовательных

		траекторий, организации системы общения между всеми участниками образовательного процесса, в том числе с использованием новых информационных технологий.
В20 Формирование навыков коммуникации, командной работы и лидерства		1.Использование воспитательного потенциала дисциплин профессионального модуля командной работы и лидерства, творческого инженерного мышления, стремления следовать инженерного мышления, стремления следовать в профессиональной деятельности нормам поведения, обеспечивающим нравственный характер трудовой деятельности и неслужебного поведения, ответственности за принятые решения через подготовку групповых курсовых работ и практических заданий, решение кейсов, прохождение практик и подготовку ВКР. 2.Использование воспитательного потенциала дисциплин профессионального модуля для: - формирования производственного коллективизма в ходе совместного решения как модельных, так и практических задач, а также путем подкрепление рационально-технологических навыков взаимодействия в проектной деятельности эмоциональным эффектом успешного взаимодействия, ощущением роста общей эффективности при распределении проектных задач в соответствии с сильными компетентностными и эмоциональными свойствами членов проектной группы.
В21 Формирование способности и стремления следовать в профессии нормам поведения, обеспечивающим нравственный характер трудовой деятельности и неслужебного поведения		
В22 Формирование творческого инженерного/профессионального мышления, навыков организации коллективной проектной деятельности		
В30 Формирование навыков цифровой гигиены		Использование воспитательного потенциала дисциплин "Информатика", "Объектно-ориентированное программирование", "Технологии программирования", "Базы данных", "Сети и телекоммуникации", "Операционные системы", "Защита информации"- для формирования
В31 Формирование ответственности за обеспечение кибербезопасности		
В32 Формирование профессионально значимых установок		

		культуры написания и оформления программ, формирование навыков цифровой гигиены, формирование ответственности за обеспечение кибербезопасности, формирование профессионально значимых установок: не работать на заказчика, чьи цели противоречат гуманистическим ценностям.
Интеллектуальное воспитание	В11 Формирование культуры умственного труда	Использование воспитательного потенциала дисциплин гуманитарного, естественнонаучного, общепрофессионального и профессионального модуля для формирования культуры умственного труда посредством вовлечения студентов в учебные исследовательские задания, курсовые работы и др.

5 СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

5.1 Структура курса «Технологии программирования»

Общая трудоемкость дисциплины составляет **4 зачетные единицы, 144 часа.**

№ п/п	Название темы/раздела учебной дисциплины	Виды учебной деятельности, включая самостоятельную ра- боту студентов и трудоемкость (в часах)				Текущий контроль (форма*, неделя)	Максимальный балл за раздел	Индикаторы освоения компетенции
		Лекции	Лабораторные занятия	Практические занятия	Самостоятельная работа			
1	Введение в программиро- вание на языке Java	8	12	18	15.5	ПКР (18) КТ (6)	33	3-ПК-10.1 У-ПК-10.1 В-ПК-10.1 3-ПК-10.3 У-ПК-10.3 В-ПК-10.3
2	Объектно- ориентированное про- граммирование	4	8	-	5	ДЗ1 (14)	16	
3	Разработка графического интерфейса пользователя	6	16	-	6.5	ДЗ2 (17)	21	
Итого:		18	36	18	27		70	
Экзамен							30	

*Сокращение наименований форм текущего контроля и аттестации разделов:

ПКР – практическая контрольная работа, ДЗ – домашнее контрольное задание.

КТ – контрольный тест

5.2 Содержание лекционных занятий (7-й семестр) – 18 часов

Неделя	Раздел курса, № занятия	Темы лекционных занятий	Трудоемкость, час.
1	Раздел 1 Л1	Эволюция технологий программирования. «Стихийное» программирование. Процедурное программирование. Структурное программирование. Модульное программирование. Объектно-ориентированное программирование. Платформа Java и язык Java. Интерфейс прикладного программирования Java (JavaAPI). Выполнение Java-программы. Приложения и апплеты. Интегрированные среды Java.	2
2	Раздел 1 Л2	Введение в программирование на языке Java. Структура Java-программы. Методы в Java. Статические переменные и методы. Переменные и методы экземпляра. Идентификаторы. Переменные класса, переменные экземпляра, локальные переменные, параметры. Типы данных, преобразование типов. Операции в языке Java, приоритет операций. Операторы, виды операторов. Организация ввода/вывода данных в консольном приложении на Java. Линейные программы. Условный оператор и оператор варианта. Разветвляющиеся программы на Java. Операторы циклов.	2
3	Раздел 1 Л3	Массив. Описание одномерного массива. Типовые алгоритмы: ввод, вывод, инициализация, сортировка массива. Алгоритмы сортировки: простого выбора, обменов, пузырьковый и пр. Двумерные и многомерные массивы.	2
4	Раздел 1 Л4	Строковый тип данных String. Методы, назначение, виды. Формальные и фактические параметры. Рекурсивные методы. Хранение данных отдельно от программы. Файлы. Виды файлов. Прямой и последовательный доступ к записям файла.	2
5	Раздел 2 Л5	Основы объектно-ориентированного программирования (ООП). Понятие и общие принципы ООП. Достоинства и недостатки ООП. Объектная декомпозиция. Классы, интерфейсы, пакеты. Объявление классов. Объявление переменных класса. Объявление методов класса. Передача аргументов по ссылке и по значению. Использование ключевых слов this и super.	2
6	Раздел 2 Л6	Наследование и управление доступом. Одиночное и множественное наследование, интерфейсы. Инкапсуляция. Скрытие данных. Полиморфизм. Композиция классов. Модификаторы доступа и области видимости членов класса.	2
7	Раздел 3 Л7	Создание графического интерфейса пользователя и апплетов. Графический интерфейс пользователя. Принципы построения. Графические библиотеки. Графические примитивы.	2
8	Раздел 3 Л8	Введение в библиотеку Swing. Обзор визуальных компонентов Swing. Текстовые компоненты и таблицы. Обработка событий. Обработка действий мыши и клавиатуры. Создание меню и диалоговые окна в Swing. Апплеты	2
9	Раздел 3 Л9	Нетривиальные возможности Java. Обработка исключений. Причины возникновения исключений. Алгоритм обработки исключений. Конструкция try-catch-finally. Оператор throw. Проверяемые и непроверяемые исключения. Создание пользовательских классов исключений	2

5.3 Темы лабораторных занятий (7-й семестр) – 36 часов

Неделя	Раздел курса, № занятия	Темы лабораторных занятий Мероприятие по текущему аудиторному контролю знаний	Трудоемкость, час.
1-2	Раздел 1 ЛР1	Основы работы в среде Eclipse. Знакомство со средой Eclipse. Редакторы, представления и компоновки. Компоновка Java. Создание простого консольного приложения. Алгоритмизация и использование управляющих структур в Java. Использование элементов организации ветвления и циклов в приложении Java.	4
3-4	Раздел 1 ЛР2	Работа с массивами. Создание приложения, работающего с одномерными и многомерными массивами.	4
5-6	Раздел 1 ЛР3, КТ	Работа со строками. Создание приложения, использующего текстовые типы данных и выполняющего основные действия над строками. КТ «Технология Java и основы языка Java»	4
7-8	Раздел 2 ЛР4	Объектно-ориентированное программирование. Создание приложения, демонстрирующего основные принципы объектно-ориентированного программирования (инкапсуляция, наследование, полиморфизм)	4
9-10	Раздел 2 ЛР5	Использование классов-коллекций. Создание приложения, использующего нетривиальную обработку большого объема информации	4
11-12	Раздел 3 ЛР6	Работа с графическим интерфейсом пользователя. Обработка событий. Создание оконного приложения из нескольких файлов	4
13-14	Раздел 3 ЛР7	Работа с графическим интерфейсом пользователя. Обработка событий. Создание приложения, использующего графические примитивы. Построение графиков функций	4
15-16	Раздел 3 ЛР8	Работа с графическим интерфейсом пользователя. Обработка событий. Создание приложения, использующего текстовые и табличные компоненты	4
17-18	Раздел 3 ЛР9	Апплеты. Создание, запуск и просмотр апплета.	4

5.4 Темы практических занятий (7-й семестр) – 18 часов

Неделя	Раздел курса, № занятия	Темы практических занятий Мероприятие по текущему аудиторному контролю знаний	Трудоемкость, час.
4-5	Раздел 1 ПР1	Линейный и бинарный поиск. Линейный поиск в неупорядоченном и упорядоченном массиве. Бинарный поиск. Поиск строки в таблице. Поиск подстроки в строке и в таблице. Алгоритм прямого поиска. Алгоритмы Кнута-Морриса-Пратта, Бойера-Мура.	4

Неделя	Раздел курса, № занятия	Темы практических занятий Мероприятие по текущему аудиторному контролю знаний	Трудоемкость, час.
8-9	Раздел 1 ПР2	Классификация алгоритмов внутренней сортировки. Сортировка подсчетом, сортировка с помощью прямого обмена, сортировка с помощью прямого выбора, сортировка с помощью прямого включения, сортировка Шелла, быстрая сортировка Хоара, пирамидальная сортировка по дереву, сортировка слиянием. Сравнение алгоритмов сортировки	4
12-13	Раздел 1 ПР3	Алгоритмы внешней сортировки: однофазная и двухфазная сортировка простым слиянием, однофазная и двухфазная сортировка естественным слиянием, многопутевое слияние, внутренняя сортировка с внешним многопутевым слиянием.	4
16-17	Раздел 1 ПР4	Нелинейные структуры данных, классификация. Деревья, деревья поиска, сбалансированные АВЛ – деревья. Использование деревьев в задачах поиска. Балансировка. Б-деревья. Цифровой поиск	4
18	Раздел 1 ПКР	Практическая контрольная работа «Алгоритмы поиска и сортировки данных»	2

5.5 Самостоятельная работа – 27 часов

Самостоятельная работа студента по учебной дисциплине регламентируется «Положением об организации самостоятельной работы студентов в НТИ НИЯУ МИФИ».

№ п/п	Виды самостоятельной работы / разделы курса	Трудоемкость, час.
1.	Изучение дополнительного материала по теме лекции	0.5 час/нед.
2.	Подготовка к лабораторным работам	0.5 час/работу
3.	Подготовка к практическим работам	2 час/работу
4.	Подготовка к практическим контрольным работам: - ПКР «Алгоритмы поиска и сортировки данных» / Раздел 1	3 час.
5.	Подготовка к компьютерному тестированию: - КТ «Технология Java и основы языка Java» / Раздел 1	1 час.
6.	Выполнение домашних заданий - ДЗ1 Графика в Java. / Раздел 2 - ДЗ2 Создание статического изображения. / Раздел 3	3 час. 3 час.

Перечень учебно-методического обеспечения самостоятельной работы студентов приведен в Приложении 1.

Методические указания для студентов по освоению дисциплины приведены в Приложении 2.

6 ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

При реализации программы дисциплины «Технологии программирования» используются различные образовательные технологии – во время аудиторных занятий (72 часа) занятия проводятся в форме лекций и лабораторных (практических) занятий.

В процессе изучения дисциплины на лекциях, которые проводятся в специализированной аудитории, используется мультимедийный проектор и заранее подготовленный демонстрационный материал.

В начале каждого семестра все желающие студенты обеспечиваются электронными версиями методических пособий, имеющихся на кафедре, по изучаемому курсу для работы дома.

На сервере кафедры организован каталог со всеми методическими пособиями, разработанными на кафедре, для возможности постоянного студенческого доступа к ним с любого компьютера во время всех видов занятий.

Самостоятельная работа студентов (27 часов) подразумевает проработку лекционного материала с использованием рекомендуемой литературы (методических пособий по курсу) для подготовки к лабораторным и контрольным работам, контрольным тестам и зачету, а также выполнение контрольных домашних заданий и самостоятельное изучение ряда тем. Виды самостоятельной работы и их трудоемкость подробнее описаны в п. 5.5.

Для повышения уровня знаний студентов по курсу «Технологии программирования» в течение семестра организуются консультации преподавателей (согласно графику консультаций кафедры АУ). Во время консультационных занятий:

- проводится объяснение непонятных для студентов разделов теоретического курса;
- разъясняются алгоритмы решения задач индивидуальных домашних заданий;
- принимаются задолженности по тестовым и контрольным работам и т.д.

Перечень учебно-методического обеспечения самостоятельной работы студентов приведен в Приложении 1.

Методические указания для студентов по освоению дисциплины приведены в Приложении 2.

Реализация компетентного подхода предусматривает широкое использование в учебном процессе активных и интерактивных форм проведения занятий, предполагающих активную обратную связь между преподавателем и студентами.

В процессе изучения дисциплины «Технологии программирования» используются интерактивные формы обучения при проведении лабораторных (практических) занятий:

- выступление студентов с докладом по теме для самостоятельного изучения;
- защита домашнего контрольного задания;
- дискуссии;
- презентации.

7 СРЕДСТВА ДЛЯ КОНТРОЛЯ И ОЦЕНКИ

В целях повышения эффективности процесса обучения студентов и стимулирования их самостоятельной работы в течение семестра используется система контроля текущей успеваемости и достижения ПР УД, включающая:

- посещение лекций;
- выполнение лабораторных работ;
- выполнение домашних контрольных работ;
- выполнение практических контрольных работ (проверка практических навыков студента);
- выполнение контрольных тестов (программированный экспресс-опрос по теоретическому материалу);
- самостоятельное изучение ряда тем.

Для оценки достижений студента используется балльно-рейтинговая система (Приложение 3).

Для целей промежуточной аттестации используется фонд оценочных средств (ФОС) по дисциплине (Приложение 4).

Результаты каждого тестового задания оцениваются в баллах, на основании которых выставляется оценка.

Задание, по которому проводится тест, считается зачтенным, если по нему набрано не менее половины от максимального количества баллов.

К экзамену в конце семестра студент допускается, если он сдал все лабораторные работы, выполнил все тестовые задания на положительные оценки, а также сдал все домашние контрольные задания.

На экзамене студенту предлагается выполнить 2 конкретных практических задания на компьютере по различным темам курса.

Итоговая экзаменационная оценка по курсу выводится с учетом балла, полученного на экзамене, и баллов, полученных по указанным выше компонентам аттестации текущей работы студента в семестре. Шкала перевода баллов в традиционную систему оценок представлена в следующей таблице:

Оценка по 5 бальной шкале	Зачет	Сумма баллов по дисциплине	Оценка (ECTS)	Градация
5 (отлично)	Зачтено	90-100	A	Отлично
4 (хорошо)		85-89	B	Очень хорошо
		75-84	C	Хорошо
		70-74	D	Удовлетворительно
		65-69		
3 (удовлетворительно)		60-64	E	Посредственно
2 (неудовлетворительно)	Не зачтено	Ниже 60	F	Неудовлетворительно

8 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

8.1 Перечень литературы, необходимой для освоения дисциплины

1. Жоголев Е.А. Технология программирования. –М., Научный мир, 2004. -216 с.
2. Иванова Г.С. Технология программирования. - М.: Изд-во МГТУ им. Н.Э.Баумана, 2002. -320 с.
3. Канер С., Фолк Д., Нгуен Е.К. Тестирование программного обеспечения. - Киев: «ДиаСофт», 2000.
4. Лахтин А.С., Исакова Л.Ю. Языки и технология программирования. Начальный курс. Учеб. пособие. Екатеринбург, 1998. -117 с.
5. Мандел Т. Разработка пользовательского интерфейса. – М.: ДМК Пресс, 2001.
6. Васильев А.Н. Java. Объектно-ориентированное программирование для магистров и бакалавров. - СПб.:Питер, 2012.-395 с.
7. Рудаков А.В. Технология разработки программных продуктов: учеб. пособие для студ. Сред. Проф. Образования . А.В. Рудаков. – 2-е изд., стер. – М.: Издательский центр «Академия», 2006. – 208 с.
8. Герберт Шилдт.Java: руководство для начинающих / Герберт Шилдт; пер. с англ. В. В. Вейтмана. –4-е изд. –М.: Вильямс, 2009. –715 с.
9. Основы программирования в JavaEclipse: Метод.указания / ОмГТУ; сост.: А. М. Минитаева, А. Н. Сухова. –Омск: Изд-во ОмГТУ, 2011. –51 с.
10. .Хорстманн К.С., Корнелл Г. Java 2. Библиотека профессионала. Т.1. Основы. – М.: Вильямс, 2012. – 816 с.
11. Вендров А.М. Проектирование программного обеспечения экономических информационных систем: Учебник. - М.: Финансы и статистика, 2000.
12. Шилдт Г. Java. Методики программирования Шилдта, – М.: И.Д. Вильямс 2008, – 512 с.
13. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ. - М.: МЦНМО, 2004.
14. БлохДж. Java. Эффективное программирование: Пер. с англ. – М.:Лори, 2002. – 224 с.

8.2 Учебно-методическое обеспечение дисциплины

1. Николаев Н.А. Введение в объектно-ориентированное программирование: Методическое пособие. Новоуральск, НГТИ, 2009, - 36 с.
2. Николаев Н.А. Оформление исходного текста программ: Методические указания для курсового проектирования. Новоуральск, НГТИ, 2006, - 44 с.
3. Тихонова Е.В. Программное обеспечение и технологии программирования Учебное пособие. Новоуральск, НГТИ, 2008. - 50 с.

8.3 Перечень ресурсов информационно-коммуникационной сети «Интернет», необходимых для освоения дисциплины

Наименование ресурса	Электронный адрес ресурса
1) Официальный сайт НТИ НИЯУ МИФИ	http://nsti.ru
2) ЭБС «Лань»	https://e.lanbook.com
3) ЭБС «IPRbooks»	https://iprbooks.ru
4) Образовательная платформа Юрайт	https://urait.ru/bcode/468952
5) Образовательный портал НИЯУ МИФИ	https://online.mephi.ru/
6) Научная библиотека НИЯУ МИФИ	http://library.mephi.ru/

9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Для осуществления образовательного процесса по дисциплине необходимо:

1 Лекционные занятия:

- аудитория, оборудованная техническими средствами для демонстрации лекций-визуализаций (проектор, экран, компьютер/ноутбук);
- комплект электронных презентаций/слайдов;

2 Лабораторные занятия:

- компьютерный класс;
- среда программирования Eclipse;

НТИ НИЯУ МИФИ располагает данными средствами в полном объеме.

Учебная дисциплина обеспечена учебно-методической документацией и материалами. Ее содержание представлено в локальной сети института и находится в режиме свободного доступа для студентов. Доступ студентов для тренинга по прохождению тестовых заданий и для самостоятельной подготовки осуществляется через компьютеры дисплейного класса (в стандартной комплектации).

В библиотечном фонде представлены необходимые учебные пособия согласно нормативам ФГОС.

Все рекомендуемые методические пособия и материалы по курсу «Технологии программирования», разработанные преподавателями кафедры, имеются в электронном виде, на бумажных носителях, представлены в УМКД. Пособия хранятся на кафедре Автоматизация управления, представлены в электронном читальном зале НТИ НИЯУ МИФИ. Электронные копии пособий также могут индивидуально предоставляться студентам по их запросу на кафедре Автоматизация управления.

Студенты своевременно обеспечиваются индивидуальными вариантами домашних заданий. Варианты заданий имеются в электронном виде и представлены в УМКД (кафедра Автоматизация управления).

Лабораторные работы по курсу осуществляются в компьютерных классах. Задания для выполнения на лабораторных работах представлены в методических пособиях кафедры.

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ

к рабочей программе по курсу
«Технологии программирования»
для ООП ВПО 09.03.01

на 20____/20____ уч.год

В рабочую программу вносятся следующие изменения:

Рабочая программа пересмотрена и одобрена на заседании кафедры «__»____20__ г.
Заведующий кафедрой АУ

на 20____/20____ уч.год

В рабочую программу вносятся следующие изменения:

Рабочая программа пересмотрена и одобрена на заседании кафедры «__»____20__ г.
Заведующий кафедрой АУ

на 20____/20____ уч.год

В рабочую программу вносятся следующие изменения:

Рабочая программа пересмотрена и одобрена на заседании кафедры «__»____20__ г.
Заведующий кафедрой АУ

Программа действительна

на 20____/20____ уч.год _____ (заведующий кафедрой АУ)

на 20____/20____ уч.год _____ (заведующий кафедрой АУ)

на 20____/20____ уч.год _____ (заведующий кафедрой АУ)

на 20____/20____ уч.год _____ (заведующий кафедрой АУ)

на 20____/20____ уч.год _____ (заведующий кафедрой АУ)

ПРИЛОЖЕНИЕ 1. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ.

№	Литература	Год	Курс	Номер группы	Семестр	Кол-во студентов	Кол-во книг	Коэффициент книгообеспеченности
Основная литература								
1	Финогенов К.Г. Основы объектно-ориентированного программирования : [лабораторный практикум]. Электронный ресурс. Москва : МИФИ. 2008 – точка доступа – ЭБС НИЯУ МИФИ – mehpri.ru	2008	4	ИТ-43	7	13	10	0,8
2	Шереметьев, А.И Информатика. Ч.1. Элементы программирования на языке С: [учебное пособие]. Электронный ресурс. Москва : НИЯУ МИФИ. 2011 – точка доступа – ЭБС НИЯУ МИФИ – mehpri.ru	2011	4	ИТ-43	7	13	10	0,8
3	Тузовский А.Ф. Объектно-ориентированное программирование. М.: Юрайт, 2018.-206 с.	2018	4	ИТ-43	7	13	10	0,8
Дополнительная литература								
1	Финогенов К.Г. Программирование в системе Windows с помощью объектно-ориентированный библиотек [лабораторный практикум]. Электронный ресурс. Москва : МИФИ. 2008 – точка доступа – ЭБС НИЯУ МИФИ – mehpri.ru	2008	4	ИТ-43	7	13	7	0,5

ПРИЛОЖЕНИЕ 2. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ СТУДЕНТОВ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ.

- стандарт организации СТО НТИ-2-2014. Требования к оформлению текстовой документации;
- методические рекомендации по организации самостоятельной работы обучающихся НТИ НИЯУ МИФИ.

ПРИЛОЖЕНИЕ 3. БАЛЛЬНО-РЕЙТИНГОВАЯ СИСТЕМА ОЦЕНКИ.

Таблица 3.1. Распределение баллов текущего рейтинга по видам деятельности студента направления подготовки 09.03.01 при изучении курса "Технологии программирования" (VII семестр)

№ п-п	Вид деятельности	Кол-во	Стоимость (в баллах)		Количество баллов	
			max	min	max	min
1.	Посещение лекций	9	1	-	9	5
2.	Выполнение лабораторных работ по расписанию	9	2	-	18	-
3.	Выполнение лабораторных работ вне расписания (без уважительных причин)	9	-	1	-	9
4.	Выполнение практических работ по расписанию	4	3		12	
5.	Выполнение практических работ вне расписания (без уважительных причин)	4		2		8
6.	Практические контрольные работы	1	5	3	5	3
7.	Выполнение домашних заданий.	2	10	5	20	10
8.	Контрольное тестирование (автоматизированное) по темам (модулям) курса:					
	- КТ - Технология Java и основы языка Java	1	6	5	6	5
Итого					70	40
9.	Экзамен	1	30		30	20
Итого					100	60

Таблица 3.2. Распределение баллов текущего рейтинга по разделам при изучении курса "Технологии программирования" студентами направления подготовки 09.03.01

№ п/п	Раздел учебной дисциплины	Виды учебной деятельности, включая самостоятельную ра- боту студентов и трудоем- кость (в часах)				Аттестация раздела (форма, неделя)	Максималь- ный балл за раздел
		Лекции	Лабораторные занятия	Практические занятия	Самостоятель- ная работа		
1	Введение в программиро- вание на языке Java	8	12	18	15.5	ПКР – (18) КТ – (6)	33
2	Объектно- ориентированное про- граммирование	4	8	-	5	ДЗ1 – (14)	16
3	Разработка графического интерфейса пользователя	6	16	-	6.5	ДЗ2 – (17)	21
4	Экзамен						30
	Итог						100

Сокращение наименований форм текущего контроля и аттестации разделов:

ДЗ – домашнее контрольное задание, КТ – контрольный тест, ПКР – практическая контрольная работа.

ПРИЛОЖЕНИЕ 4. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств по дисциплине обеспечивает проверку освоения планируемых результатов обучения (компетенций и их индикаторов) посредством мероприятий рубежного и промежуточного контроля по дисциплине.

Связь между формируемыми компетенциями и формами контроля их освоения представлена в таблице:

№ п.п.	Контролируемые модули, разделы (темы) дисциплины	Результаты освоения ООП		Виды аттестации		Наименование оценочного средства
		Код контролируемой компетенции	Индикаторы освоения компетенции	Текущий контроль	Промежуточная аттестация	
1	Введение в программирование на языке Java	ПК-10.1 ПК-10.3	3-ПК-10.1 У-ПК-10.1 В-ПК-10.1 3-ПК-10.3 У-ПК-10.3 В-ПК-10.3	ПКР КТ	По итогам текущего контроля	Компьютерное тестирование. Практическая контрольная работа
2	Объектно-ориентированное программирование			ДЗ1	По итогам текущего контроля	Домашнее задание 1
3	Разработка графического интерфейса пользователя			ДЗ2	По итогам текущего контроля	Домашнее задание 2

Варианты практических контрольных работ

ВАРИАНТ 1

1. Реализовать в программе алгоритм прямого поиска строки и КМП-алгоритм. Сравнить эффективность поиска образа в строке обоими алгоритмами по количеству итераций.
2. Реализовать в программе два алгоритма:
 - а) сортировка с помощью прямого включения,
 - б) сортировка Шелла.
 Сравнить эффективность реализованных алгоритмов по числу перестановок.

ВАРИАНТ 2

1. Реализовать в программе алгоритм прямого поиска строки и БМ-алгоритм. Сравнить эффективность поиска образа в строке обоими алгоритмами по количеству итераций.
2. Реализовать в программе два алгоритма:
 - а) сортировка с помощью двоичного включения,
 - б) шейкерная сортировка,
 Сравнить эффективность реализованных алгоритмов по числу перестановок.

ВАРИАНТ 3

1. Реализовать в программе алгоритм прямого поиска строки и КМП-алгоритм. Сравнить эффективность поиска образа в строке обоими алгоритмами по количеству итераций.
2. Реализовать в программе два алгоритма:
 - а) сортировка с помощью прямого выбора,
 - б) сортировка Шелла.
 Сравнить эффективность реализованных алгоритмов по числу перестановок.

ВАРИАНТ 4

1. Реализовать в программе алгоритм прямого поиска строки и БМ-алгоритм. Сравнить эффективность поиска образа в строке обоими алгоритмами по количеству итераций.
2. Реализовать в программе два алгоритма:
 - а) сортировка с помощью прямого включения,
 - б) шейкерная сортировка,Сравнить эффективность реализованных алгоритмов по числу перестановок.

Примерные вопросы компьютерных тестов

1. Что выдаст программа?

```
public class test {  
    public static void main (String args[]) {  
        System.out.println(6 ^ 3);  
    }  
}
```

 - 1) 243
 - 2) 5
 - 3) 18
 - 4) 0
 - 5) 6
2. Что произойдет при запуске программы?

```
public class My {  
    public static void main (String[] args) {  
        String s; System.out.println("s=" + s);  
    }  
}
```

 - 1) Программа скомпилирует и напечатает "s="
 - 2) Программа не скомпилируется, потому что строка s не инициализирована
 - 3) Программа скомпилируется, но при вызове метода toString возникнет исключительная ситуация NullPointerException
 - 4) Программа скомпилируется и напечатает "s=null"
 - 5) Программа не скомпилируется, потому что на строку s нельзя ссылаться
3. Для чего среда Java ограничивает приложениям прямой доступ к памяти?
 - 1) это требование объектной ориентированности языка Java
 - 2) для достижения высоких требований безопасности сетевых приложений
 - 3) для достижения архитектурной независимости java программ
 - 4) этим исключаются многие ошибки исполнения Java приложений еще на стадии разработки программ
4. Что необходимо поместить на место комментария в строке 4, чтобы класс скомпилировался?
 1.

```
public class ExceptionTest {
```
 2.

```
class TestException extends Exception {}
```
 3.

```
public void runTest () throws TestException {}
```
 4.

```
public void test () /* Point X*/ {
```
 5.

```
runTest ();
```
 6.

```
}
```
 7.

```
}
```
 - 1) catch (Exception e)
 - 2) throws RuntimeException
 - 3) throws Exception
 - 4) Класс скомпилируется и без изменения кода
 - 5) catch (TestException e)
5. Сколько байт оперативной памяти занимает число типа double?
 - 1) 2
 - 2) 4
 - 3) 6

- 4) 8
 - 5) Объем выделяемой памяти зависит от компилятора
 - 6) Объем выделяемой памяти зависит от интерпретатора
6. Что произойдет при попытке запуска программы?
1. class A { public byte getNumber () { return 1; }}
 2. class B extends A {
 3. public short getNumber() { return 2; }
 4. public static void main (String args[]) {
 5. B b = new B (); System.out.println(b.getNumber());
 6. }
 7. }
- 1) Программа откомпилируется и напечатает "2"
 - 2) Компилятор выдаст ошибку в строке 3
 - 3) Программа откомпилируется, но в процессе выполнения возникнет исключительная ситуация
 - 4) Компилятор выдаст ошибку в строке 5
 - 5) Программа откомпилируется и напечатает "1"
7. Каково максимальное значение для переменной типа byte?
- 1) 127
 - 2) 128
 - 3) 256
 - 4) зависит от компилятора
 - 5) зависит от интерпретатора
8. Какой метод позволяет вычислить косинус 42 градусов?
- 1) Double d = Math.cos(Math.toRadians(42));
 - 2) Double d = Math.cosine(Math.toRadians(42));
 - 3) Double d = Math.cosine(42);
 - 4) Double d = Math.cos(42);
 - 5) Double d = Math.cos(Math.toDegrees(42));
9. Какие слова являются зарезервированными словами языка Java?
- 1) switch
 - 2) throws
 - 3) throw
 - 4) implement
 - 5) super
 - 6) synchronized
10. Что произойдет при попытке запуска программы?
1. public class Test {
 2. public static void replaceJ(String text) { text.replace('j', 'I');}
 3. public static void main (String args[]) {
 4. String text = new String ("java");
 5. replaceJ(text); System.out.println(text);
 6. }
 7. }
- 1) Программа напечатает "java"
 - 2) Программа напечатает "lava"
 - 3) Компилятор выдаст ошибку в строке 4
 - 4) Компилятор выдаст ошибку в строке 2
 - 5) При выполнении возникнет исключительная ситуация
11. Сколько байт оперативной памяти занимает число типа long?
- 1) 2
 - 2) 4
 - 3) 6
 - 4) 8

5) Объем выделяемой памяти зависит от компилятора б) Объем выделяемой памяти зависит от интерпретатора

12. Какой тип должна иметь переменная i?

```
switch (i) {  
default: System.out.println("Hello");}
```

1) object

2) byte

3) float

4) double

5) long

13. Какие объявления переменных допустимы?

1) int x;

2) boolean b1=true, b2=false;

3) float y=7.0;

4) short z=1.0;

5) char c="\u0422F";

6) char c="A";

14. Что напечатает программа?

```
1. class Test { public int getLength() { return 4; } }
```

```
2. public class Sub extends Test {
```

```
3. public long getLength() { return 5; }
```

```
4. public static void main (String[] args) {
```

```
5. Test sooper = new Test();
```

```
6. Sub sub = new Sub();
```

```
7. System.out.println( sooper.getLength() + "," + sub.getLength());
```

```
8. }
```

```
9. }
```

1) 4, 4

2) 4, 5

3) 5, 4

4) 5, 5

5) Компилятор выдаст ошибку

15. Какие из перечисленных идентификаторов являются корректными?

1) abc

2) 1ab

3) _bc

4) _1c_\$ac

5) \$ac

6) for_

7) Int

8) byte

16. Какое значение переменной j напечатает программа?

```
public class My {
```

```
public static void main (String[] args)
```

```
{ int i=1, j=0; switch (i) { case 2: j+=6; case 4: j+=1; default: j+=2; case 0: j+=4; }
```

```
System.out.println("j="+j);}}
```

1) 4

2) 2

3) 1

4) 6

5) 0

17. Равны ли следующие числа:

1) 5 и 05

2) 9 и 09

3) 10 и 010

4) 0x5A и 90L

18. Какие три выражения принимают значение true.

```
public class My {  
    private int val;  
    public My (int v) { val=v; }  
    public static void main (String[] args) {  
        My a = new My(10);  
        My b = new My(10);  
        My c = a;  
        int d =10;  
        double e = 10.0;}}
```

1) d==e

2) b==d

3) a==c

4) d==10.0

5) a==b

19. Чему будет равно следующее выражение и значения переменных x и y после вычислений?

```
int x=0, y=0;  
System.out.println((++x==1) || (y++==1));
```

1) ложно, x=1, y=1

2) истинно, x=1, y=1

3) ложно, x=0, y=0

4) истинно, x=0, y=0

5) ложно, x=0, y=1

6) истинно, x=0, y=1

7) ложно, x=1, y=0 8) истинно, x=1, y=0

20. Что напечатает программа?

```
public class My {  
    public static void main (String args[]) {  
        int i=0; while (i) { if (i==4) { break; } ++i; }  
        System.out.println("i="+i);}}
```

1) 0

2) Компилятор выдаст ошибку

3) 5

4) 4

5) 3

Варианты заданий для домашних работ

Д31 Графика в Java.

1. Изучить основные понятия и термины обработки событий по модели делегирования событий на Java.

2. Разобрать приведенные примеры **FontsList**, **LinesDraw**, **KeyCodes**, **Timer**, **DrawFig**.

3. Разработать программу. Основная задача – разработка упрощенной имитации поведения объектов (все последующие лабораторные работы будут расширять это задание). Объекты реализуются через наследование: абстрактный класс, интерфейс → наследники.

Рабочий цикл программы:

- запускается процесс симуляции, генерируются объекты классов согласно заданию;
- симуляция завершается, выводится статистическая информация.

Для решения задачи:

- Создать интерфейс **IBehaviour**, задающий поведение объекта;
- Создать иерархию классов, определяющие объекты по варианту и реализующие интерфейс **IBehaviour**.

– Создать класс Habitat, определяющий размер рабочей области и хранящий список объектов, с параметрами заданными вариантом. Предусмотреть в классе метод Update, вызываемый по таймеру и получающий на вход время, прошедшее от начала симуляции. В данном методе должны генерироваться новые объекты и помещаться в поле визуализации в случайном месте. Визуализация объекта – схематично, плюсом будет, если объект будет похож на оригинал;

Рабочее окно программы – область визуализации среды обитания объектов;

4. Симуляция должна запускаться по клавише **В** и останавливаться по клавише **Е**. При остановке симуляции список уничтожается. Время симуляции должно отображаться текстом в области визуализации и скрываться/показываться по клавише **Т**;

5. По завершению симуляции в поле визуализации должна выводиться информация о количестве и типе сгенерированных объектов, а также время симуляции. Текст должен быть форматирован, т.е. выводиться с использованием разных шрифтов и цветов.

6. Параметры симуляции задаются в классе Habitat.

Вариант 1

Объект – муравей. Бывают 2 видов: рабочий и воин. Рабочие рождаются каждые N_1 секунд с вероятностью P_1 . Воины рождаются каждые N_2 секунд с вероятностью P_2 .

Вариант 2

Объект – пчела. Бывают 2 видов: трутень и рабочий. Трутни рождаются каждые N_1 секунд, если их количество менее $K\%$ от общего числа пчел, в противном случае – не рождаются вовсе. Рабочие рождаются каждые N_2 секунд с вероятностью P .

Вариант 3

Объект – аквариумная рыбка. Бывают 2 видов: золотая и гуппи. Золотые рыбки рождаются каждые N_1 секунд с вероятностью P_1 . Гуппи рождаются каждые N_2 секунд с вероятностью P_2 .

Вариант 4

Объект – кролик. Бывают 2 видов: обыкновенный и альбинос. Обыкновенные кролики рождаются каждые N_1 секунд с вероятностью P_1 . Альбиносы рождаются каждые N_2 секунд, при условии, что их количество менее $K\%$ от общего числа кроликов, в противном случае – не рождаются вовсе.

Вариант 5

Список объектов продажи на автомобильном рынке состоит из 2-х видов машин: грузовые и легковые. Грузовые машины генерируются каждые N_1 секунд с вероятностью P_1 . Легковые генерируются каждые N_2 секунд с вероятностью P_2 .

Вариант 6

Рабочий коллектив компании состоит из разработчиков и менеджеров. Разработчики генерируются каждые N_1 секунд с вероятностью P_1 . Менеджеры генерируются каждые N_2 секунд при условии, что их количество менее $K\%$ от общего числа разработчиков, в противном случае – не генерируются.

Вариант 7

Список жилых домов города состоит из двух типов: капитальный, деревянный. Капитальные дома генерируются каждые N_1 секунд с вероятностью P_1 . Деревянные дома генерируются каждые N_2 секунд с вероятностью P_2 .

Вариант 8

Список транспортных средств на дороге состоит из двух категорий: автомобили и мотоциклы. Автомобили генерируются каждые N_1 секунд с вероятностью P_1 . Мотоциклы генерируются каждые N_2 секунд с вероятностью P_2 .

Вариант 9

Объект обучающийся. Бывает 2 видов: студент (муж. пола) и студентка (жен. пола). Студенты генерируются каждые N_1 секунд с вероятностью P_1 . Студентки генерируются каждые N_2 секунд с вероятностью P_2 .

Вариант 10

Картотека налоговой инспекции состоит из записей двух типов: физические и юридические лица. Физические лица генерируются каждые N_1 секунд с вероятностью P_1 . Юридические лица генерируются каждые N_2 секунд с вероятностью P_2 .

Д32 Создание статического изображения.

Создать окно приложения с собственной панелью. На панели нарисовать статичное тематическое изображение – дом, сад, поезд, ракета и т.п.

Необходимо создать изображение с использованием разных цветов (минимум 3 разных цвета). Изображение должно быть достаточно сложным – в нем должно быть около 25–40 элементов.

Варианты:

1. Дом с садом.
2. Автомобиль на трассе
3. В кабинете программиста
4. Цветочная поляна
5. Дом
6. Новогодняя елка
7. Стол
8. Стул
9. Шкаф
10. Музыкальный инструмент
11. Молоток и наковальня
12. Принтер
13. Самолет
14. Паровоз
15. Флаги стран
16. Завод
17. Дом многоквартирный
18. Мост
19. Корабль
20. Свободная тема – можно предложить свой собственный рисунок

Задание считается выполненным при наличии бумажного листка с оцифровкой точек картинки

Перечень вопросов для подготовки к экзамену

Введение в программирование на языке Java.

1. Java-программа. Структура программы.
2. Идентификатор. Правила записи идентификатора.
3. Блок описаний.
4. Тело программы.
5. Данные, обрабатываемые программой. Виды данных: константы и переменные.
6. Стандартные типы данных. Операции над данными стандартных типов.
7. Стандартные функции и процедуры.
8. Команды присваивания, ввода, вывода.
9. Оператор ветвления в полной и сокращенной формах. Условие. Логическое выражение. Логические операции. Простое и сложное условие.
10. Составной оператор. Операторные скобки. Оператор варианта.
11. Цикл. Тело цикла. Цикл с предусловием. Цикл с постусловием. Цикл со счетчиком (с параметром). Алгоритмы вычисления суммы, произведения, среднего арифметического элементов числовой последовательности. Алгоритм поиска экстремума.
12. Система типов языка Java. Массив. Описание одномерного массива. Типовые алгоритмы: ввод, вывод, инициализация, сортировка массива. Алгоритмы сортировки массива: простого выбора, обменов, пузырьковый и пр.
13. Двумерные и многомерные массивы.
14. Строка. Операция конкатенации. Операции отношения. Стандартные функции для работы со строками. Стандартные процедуры.
15. Методы, назначение, виды.
16. Нетипизированные и типизированные методы. Формальные и фактические параметры.
17. Файлы. Виды файлов. Прямой и последовательный доступ к записям файла.

Объектно-ориентированное проектирование и программирование.

1. Класс. Поля и свойства класса.
2. Методы, их виды.
3. Объект. Имя объекта. Состояние (переменные состояния).
4. Формальное определение понятия инкапсуляция. Скрытие данных.
5. Примеры инкапсуляции в жизни и в программировании.
6. ООП как процесс построения иерархии классов. Определение наследования.
7. Общее определение полиморфизма.
8. Концепция полиморфизма. Преимущества полиморфизма. Цели полиморфизма, применительно к объектно-ориентированному программированию.
9. Модификаторы доступа к членам класса.
10. Интерфейсы и их реализация.

Разработка графического интерфейса.

1. Что такое форма? Почему первое окно формы называют “стартовым”? Что такое компонент? Как поместить компонент в окно формы?
2. Как изменить размеры и положение компонента на форме с помощью мыши? Для всех ли компонентов допускается изменение размеров? Можно ли изменить размеры и положение самой формы? Какой компонент в окне формы называют “текущим”? Может ли быть “текущим” компонентом сама форма?
3. Что такое компонент? Свойство компонента?
4. Чем отличается простое свойство от составного свойства?
5. Что такое событие? Приведите примеры известных вам событий.
6. Что такое обработчик событий? Как он оформляется?
7. Как вызвать обработчик события для нужного компонента? Для формы?
8. Перечислите известные вам события, характерные для большинства компонентов. Приведите примеры специфических событий, характерных только для формы (фреймового окна).
9. Перечислите способы ввода и вывода информации в ходе работающего приложения и опишите особенности их работы.
10. Перечислите назначение категорий всех файлов, входящих в структуру проекта.
11. Объясните назначение главного модуля проекта и опишите его структуру. Как можно просмотреть текст этого модуля? Как вы думаете, почему не рекомендуется исправлять его содержимое?
12. Перечислите правила, которые требуется соблюдать для того, чтобы файлы вашего проекта правильно сохранялись на диске, а приложение нормально работало.
13. Объясните, что означает термин “невизуальный компонент”. Приведите примеры таких компонентов.
14. Перечислите стандартные окна диалога, которые работают со списками имен файлов. Что нужно сделать для того, чтобы в этих окнах появлялись файлы заданных типов?
15. Как поместить рисунок на форму? Какие методы для этой цели существуют и в чем их принципиальное отличие?
16. Играет ли какую-либо роль последовательность помещения компонентов на форму? Можно ли изменить порядок в этой последовательности и как это сделать?
17. Какие способы вывода графической информации вам известны?
18. Что собой представляет компонент Timer и как им можно воспользоваться в разрабатываемом приложении?
19. В чем отличие модального режима работы окна от немодального?
20. Как создать в приложении новую форму?
21. Что такое Менеджер компоновки, какие бывают, их отличия.
22. Поясните назначение событий, связанных с перетаскиванием объекта мышью.

Образцы экзаменационных билетов

Национальный исследовательский ядерный университет "МИФИ"
Новоуральский технологический институт
Кафедра *автоматизации управления*

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ N 1

По курсу "Технологии программирования"
для направления подготовки 09.03.01 (очная форма обучения), VII семестр

1. Составить программу, которая выведет на экран таблицу умножения на заданное число.
2. Создать классы ТОЧКА, ОКРУЖНОСТЬ и ЭЛЛИПС. Определить конструкторы и методы печати, переноса (сдвига) фигур, вычисления расстояния до начала координат.
Создать набор (массив) фигур (точек, окружностей и эллипсов), применить все методы, найти количество эллипсов и точек, а также самую большую окружность и самую удаленную от начала координат фигуру.

Преподаватель _____ Е.В.Тихонова

Зав.кафедрой _____ П.И. Степанов

Национальный исследовательский ядерный университет "МИФИ"
Новоуральский технологический институт
Кафедра *автоматизации управления*

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ N 2

По курсу "Технологии программирования"
для направления подготовки 09.03.01 (очная форма обучения), VII семестр

1. Составить программу, которая выведет на экран все числа из заданного интервала, кратные введенному числу.
2. Объявить класс ФАКУЛЬТЕТ, имеющий название, количество студентов и позволяющий зачислять и отчислять N студентов. Определить класс УНИВЕРСИТЕТ, представляющий собой набор факультетов и предоставляющий возможность определить среднюю численность студентов на факультетах университета, самый многочисленный и самый малочисленный факультет. Продемонстрировать работу с этим классом: зачислить студентов на все факультеты, отчислить (выпустить) некоторое количество студентов, продемонстрировать работу остальных методов.

Преподаватель _____ Е.В.Тихонова

Зав.кафедрой _____ П.И. Степанов

Национальный исследовательский ядерный университет "МИФИ"
Новоуральский технологический институт
Кафедра *автоматизации управления*

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ N 3

По курсу "Технологии программирования"
для направления подготовки 09.03.01 (очная форма обучения), VII семестр

1. Составить программу, которая изобразит в нижней части экрана строку, «бегущую» справа налево. Строка должна менять свой цвет.
2. Разработать класс ФИГУРА с методами определения площади, периметра и печати (вывода информации о фигуре). Создать классы КРУГ, ПРЯМОУГОЛЬНИК, ТРЕУГОЛЬНИК.
Создать массив геометрических фигур, вычислить среднюю площадь, количество треугольников и максимальную площадь прямоугольника.

Преподаватель _____ Е.В.Тихонова

Зав.кафедрой _____ П.И. Степанов

Национальный исследовательский ядерный университет "МИФИ"

Новоуральский технологический институт

Кафедра *автоматизации управления*

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ N 4

По курсу "Технологии программирования"

для направления подготовки 09.03.01 (очная форма обучения), VII семестр

1. Составить программу, которая выведет на экран все простые числа из заданного интервала.
2. Создать класс ЖИВОТНОЕ, имеющий классификацию (строка), число конечностей, число потомков. Определить конструктор, метод печати. Создать класс ДОМАШНЕЕ ЖИВОТНОЕ, имеющее кличку. Определить методы переназначения названия клички и числа потомков. Создать набор (массив) животных, вывести информацию о них, определить количество домашних животных, продемонстрировать работу остальных методов.

Преподаватель _____ Е.В.Тихонова

Зав.кафедрой _____ П.И. Степанов