Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 09.12.2025 12:51:11
Уникальный программный ключ:
8db180d1a3f02ac9e60521a5672742735c18b1d6

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение
высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
Факультет информационных технологий

УТВЕРЖДАЮ
Декан факультета
«Информационные технологии»
_____ / Д.И. Демидов /
«__» _____ 2025 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Прикладное программирование»

Направление подготовки
09.03.03 Прикладная информатика

Профиль
«Разработка и интеграция бизнес-приложений»

Квалификация
Бакалавр

Формы обучения
очная

Москва, 2025 г.

Разработчик рабочей программы дисциплины «Прикладное программирование» профиля «Корпоративные информационные системы» направления подготовки 09.03.03 «Прикладная информатика»:

05.02.2024

старший преподаватель /Е.А. Харченко/

Согласовано:

Заведующий кафедрой «Инфокогнитивные технологии»,
доцент, к.т.н. /Е.А.Пухова/

Содержание

1 Цели, задачи и планируемые результаты обучения по дисциплине 4
2 Место дисциплины в структуре образовательной программы 5
3 Структура и содержание дисциплины 5
3.1 Виды учебной работы и трудоемкость (по формам обучения) 5
3.2 Тематический план изучения дисциплины (по формам обучения) 6
3.3 Содержание дисциплины 7
3.4 Тематика семинарских/практических и лабораторных занятий 8
3.5 Тематика курсовых проектов (курсовых работ) 10
4 Учебно-методическое и информационное обеспечение 10
4.1 Нормативные документы и ГОСТы 10
4.2 Основная литература 11
4.3 Дополнительная литература 11
4.4 Электронные образовательные ресурсы 11
4.5 Лицензионное и свободно распространяемое программное обеспечение 11
4.6 Современные профессиональные базы данных и информационные справочные системы 11
5 Материально-техническое обеспечение 12
6 Методические рекомендации 12
6.1 Методические рекомендации для преподавателя по организации обучения 12
6.2 Методические указания для обучающихся по освоению дисциплины 12
7 Фонд оценочных средств 12
7.1 Методы контроля и оценивания результатов обучения 12
7.2 Шкала и критерии оценивания результатов обучения 12
7.3 Оценочные средства 13

1 Цели, задачи и планируемые результаты обучения по дисциплине

Цели дисциплины «Основы программирования»: ознакомление студентов с методами построения программ и программирования в малом; ознакомление студентов с основами современных языков объектно-ориентированного программирования; формирование у студентов практических умений применения фундаментальных знаний в области технологии и практики современного программирования; формирование у студентов профессиональных навыков применения этих знаний.

Задачей дисциплины является обучение студентов методам построения программ и программирования в малом, выработка у студентов навыков владения современными языками объектно-ориентированного программирования, освоение студентами фундаментальных знаний в области технологии и практики современного программирования.

В результате изучения дисциплины студенты должны знать: понятия алгоритма и алгоритмического языка; базисные методы обработки информации; способы построения программ; основные концепции объектно-ориентированного программирования; базовые конструкции языка Java.

В результате изучения дисциплины студенты должны уметь: реализовывать простейшие алгоритмы обработки информации на языке Java; использовать различные типы данных; применять основные концепции объектно-ориентированного программирования.

В результате изучения дисциплины студенты должны владеть: навыками по созданию и модификации небольших программ; навыками разработки и отладки программ на алгоритмическом процедурном языке программирования высокого уровня; навыками проверки корректности и работоспособности программ.

Обучение по дисциплине «Прикладное программирование» направлено на формирование у обучающихся следующих общеобразовательных и профессиональных компетенций:

Код и наименование компетенций	Индикаторы достижения компетенции
УК-1. Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач.	ИУК-1.1. Анализирует задачу, выделяя ее базовые составляющие. ИУК-1.2. Осуществляет поиск, критически оценивает, обобщает, систематизирует и ранжирует информацию, требуемую для решения поставленной задачи. ИУК-1.3. Рассматривает и предлагает рациональные варианты решения поставленной задачи, используя системный подход, критически оценивает их достоинства и недостатки.
УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений.	ИУК-2.1.Формулирует совокупность задач в рамках поставленной цели проекта, решение которых обеспечивает ее достижение. ИУК-2.2. Определяет связи между поставленными задачами, основными компонентами проекта и ожидаемыми результатами его реализации. ИУК-2.3. Выбирает оптимальные способы планирования, распределения зон ответственности, решения задач, анализа

	результатов с учетом действующих правовых норм, имеющихся условий, ресурсов и ограничений, возможностей использования.
ОПК-7. Способен разрабатывать алгоритмы и программы, пригодные для практического применения.	ИОПК-7.1. Знает основные языки программирования, операционные системы и оболочки, современные среды разработки программного обеспечения. ИОПК-7.2. Умеет составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы, интегрировать программные модули. ИОПК-7.3. Владеет языком программирования, методами отладки и тестирования работоспособности программы.

2 Место дисциплины в структуре образовательной программы

Дисциплина «Прикладное программирование» относится к части, формируемой участниками образовательных отношений, блока Б1 «Дисциплины (модули)».

Базируется на знаниях, навыках и умениях, полученных студентами в средней школе и в результате освоения дисциплины «Основы программирования».

Приобретенные в ходе изучения дисциплины «Прикладное программирование» знания, навыки и умения в большей мере требуются для освоения дисциплин: «Основы тестирования», «Алгоритмы и структуры данных», «Шаблоны проектирования приложений», «Разработка мобильных приложений», «Проектная деятельность».

3 Структура и содержание дисциплины

Общая трудоемкость дисциплины «Прикладное программирование» составляет 3 зачетные единицы (108 часов).

3.1 Виды учебной работы и трудоемкость (по формам обучения)

3.1.1 Очная форма обучения

№ п/п	Вид учебной работы	Количество часов	Семестры
			2
1	**Аудиторные занятия**	**54**	54
	В том числе:		
1.1	Лекции	8	8
1.2	Семинарские/практические занятия	0	0
1.3	Лабораторные занятия	46	46
2	**Самостоятельная работа**	**54**	54
3	**Промежуточная аттестация**		
	Экзамен	–	**Экзамен**
	Итого:	**108**	108

3.1.2 Очно-заочная форма обучения

Не предусмотрена программой.

3.1.3 Заочная форма обучения

Не предусмотрена программой.

3.2 Тематический план изучения дисциплины (по формам обучения)

3.2.1 Очная форма обучения

№ п/п	Разделы/темы дисциплины	Трудоемкость, час					
		Всего	Аудиторная работа				Самостоятельная работа
			Лекции	Семинарские/ практические занятия	Лабораторные занятия	Практическая подготовка	
1	Раздел 1. Система контроля версий Git						
1.1	Тема 1. Система контроля версий Git	13	8	0	0	0	5
2	Раздел 2. UML. Диаграммы классов						
2.1	Тема 1. UML. Диаграммы классов	7	0	0	2	0	5
3	Раздел 3. Программирование на языке Java						
3.1	Тема 1. Реализация и использование контейнеров	8	0	0	4	0	4
3.2	Тема 2. Шаблоны методов и классов	8	0	0	4	0	4
3.3	Тема 3. Контейнеры в Java	8	0	0	4	0	4
3.4	Тема 4. Регулярные выражения	8	0	0	4	0	4
3.5	Тема 5. Проект «Рекурсивный компилятор формул»	8	0	0	4	0	4
3.6	Тема 6. Проект «Стековый компилятор формул»	8	0	0	4	0	4
3.7	Тема 7. Проект «Стековый калькулятор»	8	0	0	4	0	4
3.8	Тема 8. Модульное тестирование	8	0	0	4	0	4
3.9	Тема 9. Проект «Выпуклая оболочка»	8	0	0	4	0	4
3.10	Тема 10. Обработка событий	4	0	0	2	0	2
3.11	Тема 11. Шаблон проектирования MVC	4	0	0	2	0	2
3.12	Тема 12. Проект «Построение изображения полиэдра»	8	0	0	4	0	4
Итого		**108**	**8**	**0**	**46**	**0**	**54**

3.2.2 Очно-заочная форма обучения

Не предусмотрена программой.

3.2.3 Заочная форма обучения

Не предусмотрена программой.

3.3 Содержание дисциплины

Раздел 1. Система контроля версий Git

Тема 1. Система контроля версий Git

Работа с локальным репозиторием. Работа с удаленным репозиторием.

Раздел 2. UML. Диаграммы классов

Тема 1. UML. Диаграммы классов

Нотация UML. Основные графические элементы диаграммы классов. Примеры диаграмм классов. Чтение диаграмм классов. Разработка диаграмм классов.

Раздел 3. Программирование на языке Java

Тема 1. Реализация и использование контейнеров

Стек. Односторонняя очередь. Двустороння очередь. Односвязный список. Двусвязный список. Примеры реализаций контейнеров на основе массива.

Тема 2. Шаблоны методов и классов

Шаблоны методов. Шаблоны классов. Примеры программ.

Тема 3. Контейнеры в Java

Двусвязные списки: LinkedList, ArrayList. Множества: HashSet, LinkedHashSet, TreeSet. Словари: HashMap, LinkedHashMap, TreeMap. Примеры программ.

Тема 4. Регулярные выражения

Формальные языки. Регулярные выражения. Примеры регулярных выражений.

Тема 5. Проект «Рекурсивный компилятор формул»

Изучение исходного кода проекта. Примеры расширения функционала приложения посредством модификации.

Тема 6. Проект «Стековый компилятор формул»

Изучение исходного кода проекта. Примеры расширения функционала приложения посредством модификации.

Тема 7. Проект «Стековый калькулятор»

Изучение исходного кода проекта. Примеры расширения функционала приложения посредством модификации.

Тема 8. Модульное тестирование

Изучение исходного кода проекта. Примеры расширения функционала приложения посредством модификации.

Тема 9. Проект «Выпуклая оболочка»

Изучение исходного кода проекта. Примеры расширения функционала приложения посредством модификации.

Тема 10. Обработка событий

Обработка предусмотренных событий. Обработка произвольных событий. Примеры программ.

Тема 11. Шаблон проектирования MVC

Шаблон проектирования Model-View-Controller. Примеры программ.

Тема 12. Проект «Построение изображения полиэдра»

Изучение исходного кода проекта. Примеры расширения функционала приложения посредством модификации.

3.4 Тематика семинарских/практических и лабораторных занятий

3.4.1 Семинарские/практические занятия

Не предусмотрены программой.

3.4.2 Лабораторные занятия

Тема работы 1. UML. Диаграммы классов.

Цель работы: изучение теоретических основ построения и чтения диаграмм классов; выработка практических навыков разработки диаграмм классов при проектировании и документировании программ. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию программы и диаграммы классов. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

Тема работы 2. Контейнеры. Стек.

Цель работы: изучение теоретических и практических основ реализации стека; выработка практических навыков использования стека при разработке прикладных программ. Содержание работы: изучение примера реализации стека; модификация реализации стека; применение стека для решения практических задач. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию программы. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

Тема работы 3. Контейнеры. Односторонняя и двусторонняя очереди.

Цель работы: изучение теоретических и практических основ реализации односторонней и двусторонней очередей; выработка практических навыков использования односторонней и двусторонней очередей при разработке прикладных программ. Содержание работы: изучение примера реализации односторонней очереди; модификация реализации односторонней очереди; реализация двусторонней очереди (дека); применение дека для решения практических задач. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию программы. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

Тема работы 4. Контейнеры. Односвязный и двусвязный списки.

Цель работы: изучение теоретических и практических основ реализации односвязного и двусвязного списков; выработка практических навыков использования односвязного и двусвязного списков при разработке прикладных программ. Содержание работы: изучение примера реализации односвязного списка; модификация реализации односвязного списка; применение односвязного списка для решения практических задач; реализация двусвязного списка. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию программы. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

Тема работы 5. Шаблоны классов.

Цель работы: изучение теоретических и практических основ разработки шаблонов классов; выработка практических навыков использования шаблонов классов при разработке прикладных программ. Содержание работы: модификация шаблона класса; использование шаблона класса. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию программы. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

Тема работы 6. Контейнеры в Java.

Цель работы: изучение встроенных библиотек для работы с контейнерами; выработка практических навыков использования встроенных контейнеров при разработке прикладных программ. Содержание работы: применение встроенных контейнеров. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию программы. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

Тема работы 7. Регулярные выражения.

Цель работы: изучение теоретических основ построения регулярных выражений; выработка практических навыков использования регулярных выражений при разработке прикладных программ. Содержание работы: построение регулярных выражений. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию программы. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

Тема работы 8. Проект «Рекурсивный компилятор формул».

Цель работы: выработка практических навыков разработки сложных прикладных программ, отвечающих парадигме объектно-ориентированного программирования. Содержание работы: изучение исходного кода проекта; решение разноуровневых задач на модификацию исходного кода. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию программы. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

Тема работы 9. Проект «Стековый компилятор формул».

Цель работы: выработка практических навыков разработки сложных прикладных программ, отвечающих парадигме объектно-ориентированного программирования. Содержание работы: изучение исходного кода проекта; решение разноуровневых задач на модификацию исходного кода. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию программы. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

Тема работы 10. Проект «Стековый калькулятор».

Цель работы: выработка практических навыков разработки сложных прикладных программ, отвечающих парадигме объектно-ориентированного программирования. Содержание работы: изучение исходного кода проекта; решение разноуровневых задач на модификацию исходного кода. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию программы. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

Тема работы 11. Модульное тестирование.

Цель работы: изучение теоретических и практических основ разработки модульных тестов; выработка практических навыков использования модульных тестов при разработке прикладных программ. Содержание работы: разработка модульных тестов. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию

программы. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

Тема блока работ 12, 13. Проект «Выпуклая оболочка».

Цель работы: выработка практических навыков разработки сложных прикладных программ, отвечающих парадигме объектно-ориентированного программирования. Содержание работы: изучение исходного кода проекта; решение разноуровневых задач на модификацию исходного кода. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию программы. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

Тема блока работ 14, 15. Обработка событий и шаблон проектирования MVC.

Цель работы: изучение теоретических и практических основ приложений в концепции MVC; выработка практических навыков разработки прикладных программ в концепции MVC. Содержание работы: изучение исходного кода примера; модификация исходного кода примера; разработка приложений в концепции MVC. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию программы. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

Тема работы 16. Проект «Построение изображение полиэдра».

Цель работы: выработка практических навыков разработки сложных прикладных программ, отвечающих парадигме объектно-ориентированного программирования. Содержание работы: изучение исходного кода проекта; решение разноуровневых задач на модификацию исходного кода. Пример задания представлен в разделе 7.3. Результат работы студента: разработанные согласно заданию программы. Выполнение и защита работы направлены на формирование у студента компетенций УК-1, УК-2 и ОПК-7.

3.5 Тематика курсовых проектов (курсовых работ)

Не предусмотрены программой.

4 Учебно-методическое и информационное обеспечение

4.1 Нормативные документы и ГОСТы

1. Федеральный закон от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации» (с изменениями и дополнениями).

2. Федеральный государственный образовательный стандарт высшего образования – бакалавриат по направлению подготовки 09.03.03 «Прикладная информатика», утвержденный приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 922.

3. Приказ Министерства образования и науки РФ от 05 апреля 2017 г. № 301 «Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры».

4. «Порядок проведения государственной итоговой аттестации по образовательным программам высшего образования – программам бакалавриата, программам специалитета и программам магистратуры», утвержденный приказом Минобрнауки России от 29 июня 2015 г. № 636.

5. Положение «О практической подготовке обучающихся», утвержденное приказом Министерства науки и высшего образования Российской Федерации и Министерства просвещения Российской Федерации от 5 августа 2020 г. № 885/390.

6. Устав и локальные нормативные акты Московского политеха.

4.2 Основная литература

1. Пономарчук, Ю.В. Программирование на языке Java: учебное пособие / Ю.В. Пономарчук, И.В. Кузнецов. – Хабаровск: ДВГУПС, 2021. – 103 с. – Текст: электронный // Лань: электронно-библиотечная система. – URL: https://e.lanbook.com/book/259451 (дата обращения: 28.09.2023). – Режим доступа: для авториз. пользователей.

2. Строганкова, Н.В. Шаблоны программных платформ языка Java: учебное пособие / Н.В. Строганкова, К.В. Касьяненко, А.В. Хозяинов; составители практикума входят: использование. – Москва: РТУ МИРЭА, 2021. – 83 с. – Текст: электронный // Лань: электронно-библиотечная система. – URL: https://e.lanbook.com/book/182466 (дата обращения: 28.09.2023). – Режим доступа: для авториз. пользователей.

4.3 Дополнительная литература

1. Хорстманн К. С. Java. Библиотека профессионала. Том 1. Основы. – СПб.: ООО "Диалектика", 2020. – 864 с.

2. Шилдт Г. Java. Руководство для начинающих. – СПб.: ООО "Диалектика", 2019. – 816 с.

3. Буч Г., Рамбо Д., Якобсон И. Язык UML. Руководство пользователя. – М.: ДМК Пресс, 2008. – 496 с.

4. Эккель Б. Философия Java. – СПб.: Питер, 2018. – 1168 с.

5. Шилдт Г. Java. Полное руководство. – М.: ООО "И.Д. Вильямс", 2022. – 1344 с.

6. Лафоре Р. Структуры данных и алгоритмы в Java. – СПб.: Питер, 2018. – 704 с.

7. Роганов Е.А. Информатика: методические указания по выполнению курсовой работы для студентов направления «Прикладная математика и информатика» и специальности «Математическое обеспечение и администрирование информационных систем». – М.: МГИУ, 2007. – 32 с.

4.4 Электронные образовательные ресурсы

1. Java. Базовый курс. – URL: https://stepik.org/courses/187

2. Основы информатики и программирования. – URL: https://intuit.ru/studies/courses/105/105/info

3. Learn Git Branching. – URL: https://learngitbranching.js.org/?locale=ru_RU

Соответствующий настоящей рабочей программе дисциплины электронный образовательный ресурс находится в разработке.

4.5 Лицензионное и свободно распространяемое программное обеспечение

1. Браузер, Atom (как пример): https://browser.ru

2. Комплект программ JDK (Java Development Kit): https://www.oracle.com/java/technologies/downloads

3. Интегрированная среда разработки, Intellij IDEA Community Edition (как пример): https://www.jetbrains.com/idea/download

4. Система контроля версий Git: https://git-scm.com/downloads

5. Редактор UML-диаграмм, UMLet (как пример): https://www.umlet.com/changes.htm

4.6 Современные профессиональные базы данных и информационные справочные системы

1. Java Documentation. – URL: https://docs.oracle.com/en/java

2. Intellij IDEA Documentation. – URL: https://www.jetbrains.com/help/idea

3. Git Documentation. – URL: https://git-scm.com/book/ru/v2

5 Материально-техническое обеспечение

Требования к оборудованию в аудиториях для проведения лекций: доска, проектор. Требования к аудиториям для проведения лабораторных занятий: компьютерный класс с доступом в интернет. Необходимое для проведения лекций и лабораторных занятий программное обеспечение приведено в разделе 4.5.

6 Методические рекомендации

6.1 Методические рекомендации для преподавателя по организации обучения

Методические рекомендации по организации лабораторного занятия: 1) обзор теоретического минимума для выполнения лабораторной работы; 2) разбор алгоритмов и примеров по теме работы; 3) консультация по тематике работы; 4) предварительные проверка работы и доведение до студента ошибок или недочетов работы; 5) прием работы с вопросами по решению и заданиями на модификацию решения.

6.2 Методические указания для обучающихся по освоению дисциплины

Методические указания по освоению дисциплины: 1) предварительное изучение теоретического материла лабораторных работ с обращением к источникам из списка литературы; 2) последовательное и своевременное выполнение и защита лабораторных работ.

7 Фонд оценочных средств

7.1 Методы контроля и оценивания результатов обучения

Виды текущего контроля успеваемости студентов: проверка и прием лабораторных работ. Вид промежуточной аттестации – экзамен.

В рамках каждой лабораторной работы студенту предлагается для решения несколько практических разноуровневых задач на определенную тему. На выполнение одной лабораторной работы отводится одна неделя. Все работы подлежат защите.

В рамках экзамена студенту предлагается выполнить за ограниченное время (60 минут) практическое задание по билету, а также ответить на уточняющие и дополнительные вопросы по решению и по тематике курса. При подготовке к ответу и непосредственно при ответе студенту разрешается использовать только среду разработки приложений на языке Java.

7.2 Шкала и критерии оценивания результатов обучения

Для оценки результатов обучения используется балльно-рейтинговая система. Экзаменационная оценка определяется суммой баллов, набранных студентом за выполнение и защиту лабораторных работ в течение семестра и сдачу экзамена. Вклад лабораторных работ в итоговую оценку – до 81 балла, экзаменационных заданий – до 19 баллов. Правила перевода совокупных баллов в экзаменационную оценку следующие: сумма баллов от 0 до 40 – оценка «неудовлетворительно»; сумма баллов от 41 до 60 баллов – оценка «удовлетворительно»; сумма баллов от 61 до 80 баллов – оценка «хорошо»; сумма баллов от 81 до 100 баллов – оценка «отлично».

Первично результаты выполнения и защиты каждой лабораторной работы оцениваются в баллах по следующей шкале: 3 балла – компетенции считаются освоенными на продвинутом уровне; 2 балла – компетенции считаются освоенными на базовом уровне; 1 балл – компетенции считаются не освоенными. Полученные баллы умножаются на

коэффициент сложности: для работ 1-3, 6, 7, 11, 14 коэффициент сложности равен 1; для работ 4, 5, 8-10, 12, 16 коэффициент сложности равен 2; для работ 13, 15 коэффициент сложности равен 3. Основные задания работ выполняются исключительно на занятиях.

Результаты выполнения экзаменационного задания с учетом ответов на уточняющие и дополнительные вопросы по курсу оцениваются по шкале от 0 до 19 баллов. Детализация баллов за задание: от 16 до 19 баллов – решение и ответ не содержат ошибок или существенных недочетов; от 10 до 15 баллов – решение и ответ не содержат ошибок, но имеют недочеты; от 0 до 9 баллов – решение или ответ содержат ошибки; от 0 до 3 баллов – задание не выполнено или не даны ответы на вопросы по решению.

При повторной промежуточной аттестации необходимым условием получения студентом положительной экзаменационной оценки является получение им не менее 11 баллов за выполнение экзаменационного задания (с учетом ответов на вопросы).

Экзаменационное задание считается выполненным, если: 1) студент понимает и может объяснить прикладное назначение модифицируемой программы; 2) студент понимает и может объяснить организацию кода модифицируемой программы; 3) результаты выполнения программы соответствуют заданию билета и не содержат ошибок; 4) внесенные изменения/дополнения органичны исходной структуре кода, не искажают его; 5) в модифицируемый код не внесены изменения, не влияющие на решение задания по билету; 6) студент может объяснить и обосновать свое решение.

7.3 Оценочные средства

7.3.1 Текущий контроль

Пример задания из лабораторной работы 1.

Постройте диаграмму классов для информационной системы компании по прокату велосипедов. В системе должны храниться данные обо всех бронированиях и арендах велосипедов: 1) каждый клиент имеет номер и серию паспорта, идентификационный номер, имя и адрес; 3) каждая модель велосипеда имеет уникальное название, тип (городской, дорожный, горный) и количество передач; 4) каждый велосипед имеет уникальный идентификационный номер и модель; 5) у компании есть несколько различных магазинов, где можно забрать и вернуть велосипеды; каждый из магазинов идентифицируется уникальным именем и имеет адрес (оба обязательны); 6) при бронировании необходимо знать следующее: какой клиент сделал бронирование, когда он заберет велосипед (день), какую модель велосипеда он хочет и где он заберет велосипед (магазин); 7) ведется учет о находящихся в аренде велосипедах; 8) при возврате велосипеда сохраняется дата возврата.

Пример задания из лабораторной работы 2.

Напишите программу, которая с помощью стека проверяет правильность расстановки парных скобок в введенной строке.

Пример задания из лабораторной работы 3.

Напишите программу, которая с помощью дека определяет, является ли введенная строка палиндромом.

Пример задания из лабораторной работы 4.

Постройте ссылочную реализацию ограниченного двусвязного списка целых чисел на основе трех одномерных статических массивов.

Пример задания из лабораторной работы 5.

Реализуйте шаблон класса, описывающего стек.

Пример задания из лабораторной работы 6.

Напишите программу, реализующую телефонный справочник для абонентов. Предусмотрите возможность добавления новых абонентов, удаления абонентов, поиска фамилии абонента по номеру его телефона.

Пример задания из лабораторной работы 7.

Напишите, не используя производных операций, регулярное выражение, описывающее множество действительных чисел (без незначащих нулей).

Пример задания из лабораторной работы 8.

Модифицируйте исходный код эталонного проекта так, чтобы программа таже компилировала формулы, содержащие унарные арифметические операции + и −.

Пример задания из лабораторной работы 9.

Модифицируйте исходный код эталонного проекта так, чтобы в качестве имен переменных допускались многосимвольные идентификаторы.

Пример задания из лабораторной работы 10.

Модифицируйте исходный код эталонного проекта так, чтобы программа также вычисляла значения формул, содержащих операцию возведения в степень **. Новую операцию следует считать правоассоциативной и имеющей максимальный приоритет.

Пример задания из лабораторной работы 11.

С помощью библиотеки JUnit разработайте модульные тесты для эталонного проекта «Стековый калькулятор».

Пример задания из блока лабораторных работ 12, 13.

Модифицируйте исходный код эталонного проекта так, чтобы программа индуктивно вычисляла количество ребер выпуклой оболочки, параллельных осям координат.

Пример задания из блока лабораторных работ 14, 15.

Разработать оконное приложение, реализующее трехмерную модель куба и визуализирующее его состояние. Режимы визуализации: 1) ортогональная или перспективная проекция; 2) прозрачные или закрашенные грани.

Пример задания из лабораторной работы 16.

Модифицируйте исходный код эталонного проекта так, чтобы невидимые части ребер полиэдра изображались пунктиром.

7.3.2 Вопросы для самопроверки

Типовые вопросы по каждой лабораторной работе:

1. Опишите прикладное назначение учебного примера.
2. Опишите общую структуру исходного кода учебного примера.
3. Опишите назначение каждого класса.
4. Опишите структуру каждого класса.
5. Опишите способы взаимодействия между объектами классов.

7.3.3 Вопросы и задания к защите лабораторных работ

Вопросы к работе 1:

1. Перечислите и опишите основные элементы диаграммы классов.
2. Опишите этапы построения диаграммы классов.
3. Опишите способ чтения диаграммы классов.
4. Объясните решения заданий лабораторной работы.

Вопросы к работе 2:
1. Дайте определение стеку.
2. Объясните реализацию стека.
3. Приведите примеры использования стеков.
4. Объясните структуру разработанных программ.
5. Объясните детали реализации разработанных программ.
6. Объясните порядок выполнения разработанных программ.
7. Расширьте функционал разработанных программ.

Вопросы к работе 3:
1. Дайте определение односторонней очереди.
2. Объясните реализацию односторонней очереди.
3. Приведите примеры использования односторонней очереди.
4. Дайте определение двусторонней очереди.
5. Объясните реализацию двусторонней очереди.
6. Приведите примеры использования двусторонней очереди.
7. Объясните структуру разработанных программ.
8. Объясните детали реализации разработанных программ.
9. Объясните порядок выполнения разработанных программ.
10. Расширьте функционал разработанных программ.

Вопросы к работе 4:
1. Объясните сущность ссылочной адресации.
2. Объясните особенности ссылочной реализации односвязного списка.
3. Объясните структуру разработанных программ.
4. Объясните детали реализации разработанных программ.
5. Объясните порядок выполнения разработанных программ.
6. Расширьте функционал разработанных программ.

Вопросы к работе 5:
1. Дайте определения шаблонам метода и класса.
2. Приведите примеры шаблонов методов и классов.
3. Объясните структуру разработанных программ.
4. Объясните детали реализации разработанных программ.
5. Объясните порядок выполнения разработанных программ.
6. Расширьте функционал разработанных программ.

Вопросы к работе 6:
1. Перечислите основные встроенные контейнеры.
2. Опишите способы работы со встроенными контейнерами.
3. Объясните структуру разработанных программ.
4. Объясните детали реализации разработанных программ.
5. Объясните порядок выполнения разработанных программ.
6. Расширьте функционал разработанных программ.

Вопросы к работе 7:
1. Дайте определение регулярному выражению.
2. Приведите примеры производных регулярных выражений.
3. Объясните структуру разработанных программ.
4. Объясните детали реализации разработанных программ.
5. Объясните порядок выполнения разработанных программ.
6. Расширьте функционал разработанных программ.

Вопросы к работе 8:

1. Опишите постановку задачи, решаемую учебным проектом «Рекурсивный компилятор формул».
2. Объясните по диаграмме классов и исходному коду структуру учебного проекта «Рекурсивный компилятор формул».
3. Объясните детали реализации учебного проекта «Рекурсивный компилятор формул».
4. Объясните порядок выполнения учебного проекта «Рекурсивный компилятор формул».
5. Расширьте функционал учебного проекта «Рекурсивный компилятор формул».

Вопросы к работе 9:

1. Опишите постановку задачи, решаемую учебным проектом «Стековый компилятор формул».
2. Объясните по диаграмме классов и исходному коду структуру учебного проекта «Стековый компилятор формул».
3. Объясните детали реализации учебного проекта «Стековый компилятор формул».
4. Объясните порядок выполнения учебного проекта «Стековый компилятор формул».
5. Расширьте функционал учебного проекта «Стековый компилятор формул».

Вопросы к работе 10:

1. Опишите постановку задачи, решаемую учебным проектом «Стековый калькулятор».
2. Объясните по диаграмме классов и исходному коду структуру учебного проекта «Стековый калькулятор».
3. Объясните детали реализации учебного проекта «Стековый калькулятор».
4. Объясните порядок выполнения учебного проекта «Стековый калькулятор».
5. Расширьте функционал учебного проекта «Стековый калькулятор».

Вопросы к работе 11:

1. Дайте определение модульным тестах.
2. Опишите подходы к разработке модульных тестов.
3. Объясните структуру разработанных программ.
4. Объясните детали реализации разработанных программ.
5. Объясните порядок выполнения разработанных программ.
6. Расширьте функционал разработанных программ.

Вопросы к блоку работ 12, 13:

1. Опишите постановку задачи, решаемую учебным проектом «Выпуклая оболочка».
2. Объясните по диаграмме классов и исходному коду структуру учебного проекта «Выпуклая оболочка».
3. Объясните детали реализации учебного проекта «Выпуклая оболочка».
4. Объясните порядок выполнения учебного проекта «Выпуклая оболочка».
5. Расширьте функционал учебного проекта «Выпуклая оболочка».

Вопросы к блоку работ 14, 15:

1. Опишите подходы к обработке типовых событий.
2. Приведите пример обработки типового события.
3. Опишите подходы к обработке произвольных событий.
4. Приведите пример обработки произвольного события.
5. Опишите шаблон проектирование MVC.
6. Приведите пример программы, структура которой соответствует шаблону MVC.

7. Объясните структуру учебного проекта «Счетчик».
8. Объясните детали реализации учебного проекта «Счетчик».
9. Объясните порядок выполнения учебного проекта «Счетчик».
10. Расширьте функционал учебного проекта «Счетчик».
11. Объясните структуру разработанных программ.
12. Объясните детали реализации разработанных программ.
13. Объясните порядок выполнения разработанных программ.
14. Расширьте функционал разработанных программ.

Вопросы к работе 16:

1. Опишите постановку задачи, решаемую учебным проектом «Изображение полиэдра».
2. Объясните по диаграмме классов и исходному коду структуру учебного проекта «Изображение полиэдра».
3. Объясните детали реализации учебного проекта «Изображение полиэдра».
4. Объясните порядок выполнения учебного проекта «Изображение полиэдра».
5. Расширьте функционал учебного проекта «Изображение полиэдра».

7.3.4 Вопросы к экзамену

1. Система контроля версий Git: теоретические основы.
2. Система контроля версий Git: применение на практике.
3. UML-диаграммы классов: теоретические основы.
4. UML-диаграммы классов: применение на практике.
5. Контейнеры: теоретические основы.
6. Контейнеры: применение на практике.
7. Шаблоны методов и классов: теоретические основы.
8. Шаблоны методов и классов: применение на практике.
9. Регулярные выражения: теоретические основы.
10. Регулярные выражения: применение на практике.
11. Модульные тесты: теоретические основы.
12. Модульные тесты: применение на практике.
13. Обработка событий: теоретические основы.
14. Обработка событий: применение на практике.
15. Шаблон проектирования MVC: теоретические основы.
16. Шаблон проектирования MVC: применение на практике.
17. Проект «Рекурсивный компилятор формул»: проектирование.
18. Проект «Рекурсивный компилятор формул»: реализация.
19. Проект «Стековый компилятор формул»: проектирование.
20. Проект «Стековый компилятор формул»: реализация.
21. Проект «Стековый калькулятор»: проектирование.
22. Проект «Стековый калькулятор»: реализация.
23. Проект «Выпуклая оболочка»: проектирование.
24. Проект «Выпуклая оболочка»: реализация.
25. Проект «Изображение полиэдра»: проектирование.
26. Проект «Изображение полиэдра»: реализация.

7.3.5 Пример экзаменационного билета

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(МОСКОВСКИЙ ПОЛИТЕХ)

Факультет «Информационные технологии», кафедра «Инфокоммуникационные технологии»
Дисциплина «Прикладное программирование»
Образовательная программа 09.03.03.01 «Прикладная информатика (корпоративные информационные системы)»
Курс 1, семестр 2

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

1. Модифицируйте эталонный проект «Изображение полиэдра» так, чтобы невидимые части ребер изображались пунктиром.

   Задание считается выполненным, если: 1) студент понимает и может объяснить прикладное назначение модифицируемой программы; 2) студент понимает и может объяснить организацию кода модифицируемой программы; 3) результаты выполнения программы соответствуют заданию билета и не содержат ошибок; 4) внесенные изменения/дополнения органичны исходной структуре кода, не искажают его; 5) в модифицируемый код не внесены изменения, не влияющие на решение задания по билету; 6) студент может объяснить и обосновать свое решение.

Утверждено на заседании кафедры « » 2023 г., протокол № .

Зав. кафедрой _____/ /