Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Сибирский государственный индустриальный университет»

Кафедра прикладных информационных технологий и программирования

УТВЕРЖДАЮ
Директор института
информационных технологий и
автоматизированных систем
_____ А.В. Зимин
подпись
«____» _____ 20___ г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Программирование

09.03.02 «Информационные системы и технологии»
(направленность (профиль): «Разработка и эксплуатация информационных систем и технологий»)

Квалификация выпускника
Бакалавр

Форма обучения
Заочная форма

Срок обучения: 4 года 6 месяцев

Год начала подготовки 2025

Новокузнецк
2025

СибГИУ
Подлинник электронного документа хранится в системе электронного согласования документов СибГИУ
ДОКУМЕНТ ПОДПИСАН ЭП
Сертификат: e95695aa2c5fd4a2ad858d663abfeefbdf88f7d3
Владелец: Директор ИИТиАС Зимин Алексей Валерьевич
Действителен: с 2 апреля 2026 г. по 26 июня 2027 г.
Дата и время: 17-04-2026 09:59:44 ПРОВЕРИТЬ ПОДПИСЬ

1 Цели и задачи освоения учебной дисциплины

Целями учебной дисциплины являются:
– получение расширенных сведений о методах разработки программ;
– совершенствование навыков программирования на одном из языков программирования высокого уровня.

Задачами учебной дисциплины являются:
– изучение основ объектно-ориентированного программирования на языке Java;
– изучение принципов разработки программ, осуществляющих контролируемый диалог с пользователем;
– получение навыков создания программ среднего уровня сложности.

2 Место учебной дисциплины в структуре ООП по направлению подготовки (специальности)

Учебная дисциплина относится к учебным дисциплинам обязательной части **Блока 1 «Дисциплины (модули)»** ООП по направлению подготовки (специальности) 09.03.02 «Информационные системы и технологии».

Учебная дисциплина базируется на предварительном усвоении обучающимися учебных дисциплин:
– Основы программирования.

Учебная дисциплина дополняет знания и умения, получаемые по одновременно изучаемым и последующим учебным дисциплинам:
– Технологии программирования.

3 Планируемые результаты обучения по учебной дисциплине

Процесс изучения учебной дисциплины направлен на формирование следующих компетенций:

– Общепрофессиональные компетенции

Наименование категории (группы) ОПК	Код и наименование ОПК	Код и наименование индикатора достижения ОПК	Планируемые результаты обучения
	ОПК-6: Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и	ОПК-6.1 Понимает теоретические основы алгоритмизации и программирования	– знать: основные элементы технологии объектно-ориентированного программирования. – уметь: разрабатывать структуру классов для конкретной задачи.
		ОПК-6.2	– знать: общие

	технологий	Разрабатывает алгоритмическое обеспечение	принципы взаимодействия пользователя с программным средством. – уметь: разрабатывать алгоритм взаимодействия пользователя с программным обеспечением.
		ОПК-6.3 Разрабатывает программное обеспечение	– знать: возможности среды разработки для организации взаимодействия пользователя с программным обеспечением. – уметь: разрабатывать программы среднего уровня сложности.

4 Объем и содержание учебной дисциплины

Учебные занятия по учебной дисциплине проводятся в форме контактной работы и в форме самостоятельной работы обучающихся.

Контактная работа включает в себя занятия лекционного типа (лекции), занятия семинарского типа (лабораторные работы, коллоквиумы), промежуточную аттестацию обучающихся и иные формы взаимодействия обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации ООП на иных условиях, в том числе при проведении промежуточной аттестации обучающихся. Контактная работа может проводиться с применением электронного обучения, дистанционных образовательных технологий.

Объем учебной дисциплины

Сессия / курс		***ИТОГО***	***2 сессия / 3 курс***	***3 сессия / 3 курс***
Форма промежуточной аттестации				*экзамен*
Трудоёмкость	*академ. час.*	***144***	*36*	*108*
	зачетных единиц	***4***	*1*	*3*
Лекции, *академ. час.*		***2***	*2*	*0*
в форме практической подготовки		***0***	*0*	*0*
Лабораторные работы, *академ. час.*		***6***	*0*	*6*
в форме практической подготовки		***0***	*0*	*0*
Практические занятия, *академ. час.*		***0***	*0*	*0*
в форме практической подготовки		***0***	*0*	*0*

Курсовая работа / проект, *академ. час.*	***0***	*0*	*0*
в форме практической подготовки	***0***	*0*	*0*
Консультации, *академ. час.*	***0***	*0*	*0*
в форме практической подготовки	***0***	*0*	*0*
Самостоятельная работа, *академ. час.*	***127***	*34*	*93*
в форме практической подготовки	***0***	*0*	*0*
Контроль, *академ. час.*	***9***	*0*	*9*
в форме практической подготовки	***0***	*0*	*0*

Содержание учебной дисциплины

Раздел 1 Реализация диалога с пользователем;

Тема 1.1 Использование компонентов выбора (Лабораторная работа, предусматривающая разработку программы с использованием компонентов выбора типа «один из нескольких» и «несколько из нескольких»);

Тема 1.2 Обработка исключений (Понятие исключения, класс Throwable и его потомки, операторы try, catch, finally, throw);

Тема 1.3 Разработка многооконных программ (Лабораторная работа, предусматривающая создание в одной программе нескольких форм с контролем правильности ввода данных, и их взаимодействие);

Тема 1.4 Использование дополнительных элементов пользовательского интерфейса (Лабораторная работа, предусматривающая использование меню, панелей инструментов, строки состояния, стандартных диалоговых окон);

Раздел 2 Создание и использование классов;

Тема 2.1 Общая структура класса (Общая схема класса, спецификаторы доступа, понятие и синтаксис полей и методов);

Тема 2.2 Конструкторы, инициализаторы объектов, перегрузка методов (Конструкторы, инициализаторы объектов, перегрузка методов класса, особенности использования параметров методов);

Тема 2.3 Доступ к полям с помощью методов (Правила организации доступа к закрытым полям);

Тема 2.4 Статические классы (Назначение статических классов и членов классов, синтаксис, особенности использования);

Тема 2.5 Наследование и полиморфизм (Понятие наследования, особенности наследования конструкторов, переопределение методов в классах-потомках. Понятие полиморфизма, назначение и синтаксис виртуальных методов, абстрактные классы, операторы приведения типов для классов, ограничение в наследовании методов и классов);

Раздел 3 Интерфейсы (Понятие интерфейса, синтаксис, особенности и примеры использования).

5 Перечень тем лекций

№ раздела / темы дисциплины	Темы лекций	Трудоемкость, *академ.час*	
		всего	**в форме практической подготовки**
Тема 1.2.	Обработка исключений	1	
Тема 2.1.	Общая структура класса	1	
Итого:		**2**	**0**

6 Перечень тем практических занятий (семинаров)

№ раздела / темы дисциплины	Темы практических занятий (семинаров)	Трудоемкость, *академ.час*	
		всего	**в форме практической подготовки**
	Отсутствуют		
Итого:		**0**	**0**

7 Перечень тем лабораторных работ

№ раздела / темы дисциплины	Темы лабораторных работ	Трудоемкость, *академ.час*	
		всего	**в форме практической подготовки**
Тема 1.1.	Компоненты выбора	2	
Тема 1.2; Тема 1.3; Тема 2.1.	Многооконные программы	2	
Тема 1.4.	Дополнительные элементы пользовательского интерфейса	2	
Итого:		**6**	**0**

8 Перечень тем курсовых работ (проектов)

№ раздела / темы дисциплины	Темы курсовых работ (проектов)	Трудоемкость, *академ.час*	
		всего	**в форме практической подготовки**
	Отсутствуют		
Итого:		**0**	**0**

9 Виды самостоятельной работы

№ раздела / темы дисциплины	Виды самостоятельной работы	Трудоемкость, *академ.час*	
		всего	**в форме практической подготовки**
Раздел 1.	1. Выполнение домашнего задания; 2. Изучение лекционного материала;	39	

	3. Изучение теоретического материала; 4. Контрольная работа; 5. Подготовка к лабораторной работе.		
Раздел 2.	1. Выполнение домашнего задания; 2. Изучение лекционного материала; 3. Изучение теоретического материала; 4. Контрольная работа; 5. Подготовка к лабораторной работе.	79	
Раздел 3.	1. Выполнение домашнего задания; 2. Изучение теоретического материала.	9	
Контроль	*Подготовка к экзамену*	9	
Итого:		**136**	**0**

10 Матрица компетентностных задач по учебной дисциплине

Контролируемые разделы (темы) дисциплины	Контролируемые компетенции	Оценочные средства
Раздел 1.	ОПК-6	1. Домашнее задание; 2. Контрольная работа; 3. Наблюдение за работой на лабораторной работе
Раздел 2.	ОПК-6	1. Домашнее задание; 2. Контрольная работа; 3. Наблюдение за работой на лабораторной работе
Раздел 3.	ОПК-6	1. Домашнее задание
Экзамен.	ОПК-6	1. Экзамен на основе результатов компьютерного тестирования

11 Входной контроль

11.1 Цель входного контроля − определить начальный уровень подготовленности обучающихся и выстроить индивидуальную траекторию обучения. В условиях личностно-ориентированной образовательной среды результаты входного оценивания обучающегося используются как начальные значения в индивидуальном профиле академической успешности обучающегося.

11.2 Описание оценочных средств

Форма проведения входного контроля – решение практической задачи.

Длительность – 30 минут. Проводится во время первой лабораторной работы.

Шкала оценивания результатов тестирования

Оценка формируется в соответствии с критериями таблицы:

Критерий оценивания	Оценка
задача решена правильно	зачтено
решение задачи дает неправильный ответ	не зачтено

Основываясь на результатах тестирования, формируется индивидуальная траектория обучающегося (группы обучающихся), исходя из следующих соображений:

– зачтено – обучающийся готов к изучению учебной дисциплины без дополнительного повторения изученного ранее материала;

– не зачтено – обучающийся имеет слабую подготовку к изучению учебной дисциплины, необходимы корректирующие занятия и консультации.

Тестовые задания для входного контроля знаний обучающихся приведены в приложении.

12 Текущий контроль

12.1 Текущий контроль знаний используется для оперативного и регулярного управления учебной деятельностью (в том числе самостоятельной) обучающихся. Данный вид контроля стимулирует у обучающихся стремление к систематической самостоятельной работе по изучению учебной дисциплины и позволяет оценить степень освоения изученного материала.

Текущий контроль успеваемости осуществляется в течение семестра, в ходе повседневной учебной работы по индивидуальной инициативе педагогического работника, в том числе посредством использования СУО СибГИУ.

Формы проведения текущего контроля – домашнее задание; наблюдение за работой на лабораторной работе; тестовые задания.

12.2 Описание оценочных средств

Шкалы оценивания (методики оценки)

12.2.1 Оценивание домашнего задания

Домашнее задание – это самостоятельная форма учебной деятельности обучающегося, которая осуществляется без непосредственного руководства и контроля со стороны педагогического работника, но по его поручению.

Критерии оценки:

– правильность ответа по содержанию задания (учитывается количество и характер ошибок при ответе);

– полнота и глубина ответа (учитывается количество усвоенных фактов, понятий и т.п.);

– сознательность ответа (учитывается понимание излагаемого материала);

– логика изложения материала (учитывается умение строить целостный, последовательный ответ, грамотно пользоваться специальной терминологией);

– рациональность использованных приемов и способов решения поставленной учебной задачи (учитывается умение использовать наиболее прогрессивные и эффективные способы достижения цели);

– использование при ответе требований нормативных документов, учебной литературы (обязательное условие);

– использование графических элементов с целью выделения особо значимой информации: таблиц, схем, рисунков и т.п.

Критерии оценки домашнего задания

Наименование параметра оценки	Оценка
Достигнут правильный результат, задание выполнено самостоятельно и оптимальным способом	зачтено
Получен неправильный результат, или задание выполнено не самостоятельно, или задание выполнено неоптимальным способом	не зачтено

12.2.2 Оценивание в процессе наблюдения за работой на лабораторном занятии

Наблюдение за работой на лабораторном занятии – важнейший компонент контроля деятельности обучающихся и качества их обучения.

Критерии оценки в процессе наблюдения за работой на лабораторном занятии:

– самостоятельность выполнения задания и работы с методическими указаниями (учитывается работа в течение занятия, быстрота и способность нахождения в методических указаниях нужной информации);

– правильность выполнения работы (учитывается последовательность выполняемых действий, умение работать с инструментами и/или средствами вычислительной техники);

– рациональность использования времени, отведенного на задание (не одобряется затянутость выполнения работы с учетом индивидуальных особенностей обучающихся);

– активность обучающегося при выполнении лабораторной работы (учитывается индивидуальная работа и работа в группе).

Критерии оценки по результатам наблюдения за работой на лабораторном занятии

Критерий оценивания	Оценка
Достигнут правильный результат, работа выполнена самостоятельно и оптимальным способом	зачтено
Получен неправильный результат, или работа выполнена не самостоятельно, или работа выполнена неоптимальным способом	не зачтено

Задания и вопросы для текущего контроля знаний обучающихся по разделам (темам) учебной дисциплины приведены в приложении.

13 Промежуточная аттестация

13.1 Оценочные материалы для промежуточной аттестации обучающихся по учебной дисциплине предназначены для оценки степени достижения запланированных результатов обучения по завершению изучения учебной дисциплины в установленной учебным планом форме и позволяют определить качество усвоения изученного материала.

Итоговой формой контроля сформированности компетенций у обучающихся по учебной дисциплине является: экзамен.

Оценочные материалы для промежуточной аттестации обучающихся включают: экзамен на основе результатов компьютерного тестирования.

13.2 Оценивание обучающегося на экзамене

Максимальный процент ответа	Проходной процент ответа	Оценка
100%	не менее 80%	отлично
79,99%	не менее 60%	хорошо
59,99%	не менее 35%	удовлетворительно
34,99%	–	неудовлетворительно

Задания и вопросы для промежуточной аттестации обучающихся приведены в приложении.

14 Учебно-методическое и информационное обеспечение учебной дисциплины

а) литература:

1 Курбатова, И. В. Основы программирования на языке Java : учебное пособие для вузов / И. В. Курбатова, А. В. Печкуров. — 2-е изд., стер. — Санкт-Петербург : Лань, 2026. — 348 с. — ISBN 978-5-50751392-5. – URL: https://e.lanbook.com/book/511511 (дата обращения: 24.03.2026);

2 Зорина, Н. В. Программирование на языке Джава : учебное пособие / Н. В. Зорина. — Москва : РТУ МИРЭА, 2021 — Часть 1 — 2021. — 164 с. – URL: https://e.lanbook.com/book/218402 (дата обращения: 24.03.2026);

3 Зорина, Н. В. Программирование на языке Джава : учебное пособие / Н. В. Зорина. — Москва : РТУ МИРЭА, 2023 — Часть 2 — 2023. — 82 с. — ISBN 978-5-7339-1765-8. – URL: https://e.lanbook.com/book/368972 (дата обращения: 24.03.2026).

б) ресурсы информационно-телекоммуникационной сети «Интернет»:

1 Электронный каталог : сайт / Научно-техническая библиотека СибГИУ. – Новокузнецк, [199 –]. – URL: http://libr.sibsiu.ru. – URL: https://libr.sibsiu.ru;

2 Электронная библиотека // Научно-техническая библиотека СибГИУ : сайт. – Новокузнецк, [200 –]. – URL: http://library.sibsiu.ru/LibrELibraryFullText.asp. – Режим доступа: для авторизир. пользователей. – URL: https://library.sibsiu.ru/LibrELibraryFullText.asp;

3 Консультант студента : электронно-библиотечная система / ООО «КОНСУЛЬТАНТ СТУДЕНТА». – Москва, [200 –]. – URL: http://www.studentlibrary.ru. – Режим доступа: для авторизир. пользователей;

4 ЛАНЬ : электронно-библиотечная система : [коллекция «Единая профессиональная база знаний для технических вузов - Издательство Лань" ЭБС»] / ООО «Издательство ЛАНЬ». – Санкт-Петербург, [200 –]. – URL: http://e.lanbook.com. – Режим доступа: для авторизир. пользователей;

5 Образовательная платформа ЮРАЙТ / ООО «Электронное издательство ЮРАЙТ». – Москва, [200 –]. – URL: https://urait.ru. – Режим доступа: для авторизир. пользователей;

6 Университетская библиотека онлайн : электронно-библиотечная система / ООО «Директ-Медиа». – Москва, [200 –]. – URL: https://biblioclub.ru. – Режим доступа: для авторизир. пользователей. – URL: http://www.biblioclub.ru;

7 НАУЧНАЯ ЭЛЕКТРОННАЯ БИБЛИОТЕКА eLIBRARY.RU : база данных / ООО «НЭБ». – Москва, [200 –]. – URL: http://elibrary.ru. – Режим доступа: по подписке;

8 Электронные периодические издания ИВИС : универсальная база данных / ООО «ИВИС». – Москва, [200 –]. – URL: http://eivis.ru. – Режим доступа: по подписке.

в) лицензионное и свободно распространяемое программное обеспечение:

– 7-Zip;
– Adobe Acrobat Reader;
– Astra Linux Special Edition;
– Eclipse;
– Kaspersky Endpoint Security;
– Microsoft Office;
– Microsoft Windows;
– OnlyOffice;
– Р7-Офис.

г) базы данных и информационно-справочные системы:

1 ГАРАНТ : справочно-правовая система / ООО «Правовой центр «Гарант». – Кемерово, [200 –]. – Режим доступа: компьютерная сеть Сиб. гос. индустр. ун-та.;

2 КонсультантПлюс : справочно-правовая система / ООО «Информационный центр АНВИК». – Новокузнецк, [199 –]. – Режим доступа: компьютерная сеть библиотеки Сиб. гос. индустр. ун-та.;

3 Техэксперт : информационно-справочная система / ООО «Группа компаний «Кодекс». – Кемерово, [200 –]. – Режим доступа: компьютерная сеть Сиб. гос. индустр. ун-та.

15 Материально-техническое обеспечение учебной дисциплины

Материально-техническое обеспечение учебной дисциплины включает учебные аудитории, оснащенные оборудованием, компьютерной техникой, и техническими средствами обучения, в том числе:

- учебную аудиторию для проведения занятий лекционного типа, оборудованную учебной доской, экраном и мультимедийным проектором;
- учебную аудиторию для проведения занятий семинарского типа (лабораторных работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенную компьютерной техникой;
- помещение для самостоятельной работы, оснащенное компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду, научно-техническую библиотеку СибГИУ.

Рабочая программа дисциплины составлена в соответствии с требованиями ФГОС ВО по направлению подготовки (специальности) 09.03.02 «Информационные системы и технологии».

Составитель(и):

доцент Кожемяченко Вадим Иванович (кафедра прикладных информационных технологий и программирования).

Рабочая программа дисциплины рассмотрена и утверждена на заседании кафедры.

Согласована:

Ген. директор ООО «Агит-плюс»		*А.Ю. Рыжих*
должность, степень, звание представителя работодателя		инициалы, фамилия

Приложение

1 Тестовое задание для входного контроля знаний обучающихся

Имеется массив, состоящий из n целых чисел. Определить индексы (позиции) всех максимальных элемента массива. Предусмотреть ввод данных с использованием компонентов классов JTable, JSpinner и вывод результата в компонент класса JTextField. Пример: для массива [10, 22, 56, 17, 41, 67, 14, 43, 67, 13] результатом будут числа 6 и 9.

2 Выполнение лабораторных работ

Лабораторная работа 1: Компоненты выбора

Пример задания лабораторной работы:

Дан одномерный массив размера *n*. Составить программу, позволяющую находить индекс первого или последнего элемента, относящегося к заданной группе. Группу элементов формировать из положительных, отрицательных и нулевых элементов в любой комбинации.

Лабораторная работа 2: Многооконные программы

Пример задания лабораторной работы: В соответствии с условиями задачи обеспечить ввод и редактирование элементов списка с использованием диалогового окна, а также удаление элементов списка. Использовать для отображения компонент класса ListBox, в котором приводить значение одного из полей элемента списка. В диалоговом окне для ввода данных использовать **только** компоненты класса TextBox (допускается использование компонентов выбора, если выбор осуществляется не более чем из пяти значений). В диалоговом окне предусмотреть контроль правильности ввода данных.

Элементом списка является анкета школьника, включающая фамилию, номер школы (1-200), номер класса (1-11) и результат теста (2-5).

Лабораторная работа 3: Дополнительные элементы пользовательского интерфейса

Пример задания лабораторной работы: Дополнить программу работы 2 следующими возможностями: сохранение введенных данных в текстовый файл и загрузку ранее сохраненного списка; выполнение над введенными данными операции, указанной ниже в задании; создание «меню», «панели инструментов»; обеспечение запрета всех недопустимых в данный момент действий; при перемещении по списку данных отображение полей элемента списка на панелях «строки состояния».

Вывести данные о школьниках в порядке убывания результата теста в виде списка строк формата «<результат теста> <фамилия> <номер школы> <номер класса>».

3 Задания и вопросы для оценки обучающегося на экзамене

Раздел 1 Реализация диалога с пользователем

Вопрос 1:

В программе имеется компонент All_CB класса JCheckBox, при выборе которого должны суммироваться все элементы целочисленного массива mas независимо от их значения.

```
int sum = 0;
for (int i = 0; i < mas.length; i++)
  if (mas[i] > 0 <...>)
    sum += mas[i];
```

Какой код, подставленный вместо <...>, позволит это реализовать?

Примечания:

- между операторами сравнения, логическими операторами и сравниваемыми величинами ставить ровно по одному пробелу (даже если возможна запись без пробела);
- использовать укороченные логические операторы;
- не использовать скобки;
- логические величины не сравнивать с true или false.

Метод ответа – ввод ответа.

Вопрос 2:

Имеется фрагмент программы:

```
if (radioButton1.Checked || ((mas[i] % 2 == 0) &&
      checkBox1.Checked))
  mas[i] += i;
```

В каком случае произойдет изменение значения элемента массива?

Метод ответа – выбор только одного ответа.

- если выбран radioButton1 или элемент кратен 2 и выбран checkBox1
- если выбран radioButton1 и элемент кратен 2 и выбран checkBox1
- если выбран radioButton1 или элемент не кратен 2 и выбран checkBox1
- если не выбран radioButton1 или элемент не кратен 2 и не выбран checkBox1

Вопрос 3:

Имеется фрагмент программы:

```
int i = 0, j = 3;
try
{
  j = Integer.parseInt("12,3") / i;
}
catch (NumberFormatException E)
{
  j = 4;
}
catch (ArithmeticException E)
{
  j = 5;
}
```

Каким будет значение переменной j после выполнения фрагмента?

Метод ответа – ввод ответа.

Вопрос 4:

Имеется фрагмент программы:

```
int i=1, j=0;
try
{
  try
  {
    i = i/j;
  }
  catch (Exception)
  {
    i = 7;
    throw;
  }
}
catch (Exception)
{
  i = 8;
}
```

Какое значение будет в переменной i после выполнения фрагмента?

Метод ответа – ввод ответа.

Вопрос 5:

Каково назначение оператора throw?

Метод ответа – выбор только одного ответа.

- это оператор генерации исключения

– создание фрагмента кода, выполняемого даже при наличии исключений
– обработка исключений
– это оператор выбора

Вопрос 6:

Имеется фрагмент программы:

```
int i = 1, j = 1;
try
{
  if (i == 0)
    throw new Exception("ArithmeticException");
  else
    throw new NumberFormatException("Exception");
}
catch (NumberFormatException E)
{
  j = 3;
}
catch (ArithmeticException E)
{
  j = 4;
}
catch (Exception E)
{
  j = 5;
}
```

Каким будет значение переменной j после выполнения фрагмента?

Метод ответа – выбор только одного ответа.

– 3
– 1
– 5
– 4
– 0

Вопрос 7:

Какое свойство формы надо использовать, чтобы при запуске она отобразилась в модальном режиме?

Метод ответа – ввод ответа.

Вопрос 8:

Какое свойство визуального компонента (например, кнопки) делает его недоступным для выбора мышью или клавиатурой, но оставляет видимым?

Метод ответа – выбор только одного ответа.

- enabled
- disabled
- closed
- unselected

Раздел 2 Создание и использование классов

Вопрос 1:

Имеются фрагменты программы:

```
public class MyClass
{
  int a;
}
```

Какой тип доступа имеет поле a?

Метод ответа – выбор только одного ответа.

- private
- protected
- public
- модификатор по умолчанию
- в описании поля имеется синтаксическая ошибка, т.к. явно не указан модификатор доступа

Вопрос 2:

Имеется описание класса:

```
public class MyClass
{
  public int a;
}
```

и фрагмент программы

```
MyClass cl = <...>;
```

Какой код, написанный вместо <...> позволит создать объект cl?

Примечание: между операторами ставить ровно по одному пробелу.

Метод ответа – ввод ответа.

Вопрос 3:

Имеется фрагмент описания класса:

```
public class MyClass
{
  public int n = 11;
```

```
  public MyClass(int m) {n = m;}
}
```

и фрагмент описания и создания объекта

```
MyClass cl = new MyClass(33) {n = 12};
```

Каким будет значение поля n объекта cl?

Метод ответа – ввод ответа.

Вопрос 4:

Имеются фрагменты программы:

```
public class MyClass
{
  <...>
}
```

Какие строки, подставленные вместо <...>, правильно описывают конструктор класса?

Метод ответа – выбор одного или нескольких ответов.

- `public MyClass() {...}`
- `public MyClass(int n) {...}`
- `public void MyClass(MyClass n) {...}`
- `public int MyClass() {...}`
- `public ~MyClass() {...}`

Вопрос 5:

Имеется фрагмент программы:

```
public class MyClass
{
  private string s;
}
```

Что нужно сделать, чтобы обеспечить извне класса доступ к полю s, но при этом иметь возможность блокировать присвоение неправильных значений?

Метод ответа – выбор только одного ответа.

- реализовать «геттер» и «сеттер»
- сделать свойство, связанное с полем
- изменить модификатор доступа на protected
- изменить модификатор доступа на public
- такое сделать невозможно

Вопрос 6:

Имеется фрагмент программы:

```
public class MyClass
{
  private int n;

  public int getN()
  {
    return value;
  }

  public void setN(int value)
  {
    n = value;
  }
}
```

Считается, что методы getN() и setN() – это «геттер» и «сеттер» для поля n.

Имеется ли ошибка в реализации методов класса?

Метод ответа – выбор только одного ответа.

- да, метод getN() должен возвращать значение n
- нет
- да, value не присвоено значение
- да, метод setN() не может обратиться к полю со спецификатором доступа private
- да, метод setN() должен быть без параметров

Вопрос 7:

Имеется фрагмент программы:

```
public class MyClass
{
  private int n;

  public int getN()
  {
    return n;
  }

  public void setN(int value)
  {
    return value;
  }
}
```

Считается, что методы getN() и setN() – это «геттер» и «сеттер» для поля n.

Имеется ли ошибка(и) в реализации методов класса?

Метод ответа – выбор одного или нескольких ответов.

- да, в методе setN() недопустимо использование оператора return с указанием возвращаемого значения
- да, метод setN() должен присваивать значение value полю n
- нет
- да, метод setN() не может иметь спецификатор доступа
- да, метод getN() не может обращаться к полям класса

Вопрос 8:

Имеется фрагмент программы:

```
public class MyClass
{
  private int n;

  public int <...>()
  {
    return n;
  }
}
```

Какое имя должно быть указано вместо <...>, если считать, что этот метод частью пары «геттер»/«сеттер» для работы с полем n?

Метод ответа – ввод ответа.

Вопрос 9:

Имеется фрагмент программы:

```
public class MyClass
{
  private int[] mas;
  public void setMas(<...>)
  {
    ...
  }
}
```

Какие параметры должны быть указаны вместо <...>, чтобы метод являлся «сеттером» поля mas?

Метод ответа – выбор одного или нескольких ответов.

- индекс редактируемого элемента
- новое значение редактируемого элемента
- старое значение редактируемого элемента

– ссылка на массив, в котором расположен редактируемый элемент

– никаких параметров не требуется

Вопрос 10:

Имеется фрагмент описания класса:

```
public class MyClass
{
  public static int n = 0;
  public MyClass()
  {
    if (n<2)
      n++;
  }
}
```

и фрагмент описания и создания объекта

```
MyClass cl;
for (int i=0; i<5; i++)
  cl = new MyClass();
```

Какое значение будет иметь поле n класса MyClass после выполнения фрагмента?

Метод ответа – ввод ответа.

Вопрос 11:

Имеется фрагмент описания класса:

```
public class MyClass
{
  public static int n = 1;
  public MyClass()
  {
    n *= 2;
  }
}
```

и фрагмент описания и создания объектов

```
MyClass cl1 = new MyClass();
MyClass cl2 = new MyClass();
int m = <...>;
```

Какие фрагменты кода, поставленные вместо <...>, позволят присвоить m значение поля n класса MyClass?

Метод ответа – выбор одного или нескольких ответов.

- MyClass.n
- cl1.n
- cl2.n
- n
- (MyClass).n

Вопрос 12:

Имеется фрагмент описания класса:

```
public static class MyClass
{
  public static int n = 1;
  public static int m = 2;
  public static void addNToM()
  {
    m += n;
  }
}
```

и фрагмент его использования:

```
MyClass.addNToM();
```

Известно, что класс MyClass не является внутренним для какого-то другого класса.

Имеется ли ошибка в данных фрагментах?

Метод ответа – выбор только одного ответа.

- да, т.к. статическим может быть только внутренний класс
- нет
- да, т.к. обращение к методу должно осуществляться через экземпляр класса, а не через класс
- да, т.к. статический класс не может иметь методов
- да, т.к. класс не может быть описан с модификатором static

Вопрос 13:

Имеется фрагмент описания класса:

```
public class MyClass
{
  public static int z = 50;
  public MyClass()
  {
    z -= 1;
```

```
  }
}
```

и фрагмент его использования

```
int m = MyClass.z;
MyClass cl = new MyClass();
```

Какое значение будет иметь m после выполнения фрагмента?

Метод ответа – ввод ответа.

Вопрос 14:

Имеется фрагмент программы:

```
public class Class1
{
  private int x = 1;
  protected int y = 1;
  public void z()
  {
    x = y = 2;
  }
}
public class Class2 extends Class1
{
  private int n = 1;
  public void m()
  {
    <...>
  }
}
```

Какие из приведенных ниже строк, подставленных вместо <...> синтаксически неправильные?

Метод ответа – выбор одного или нескольких ответов.

- x = 3;
- z();
- y = 3;
- m();
- n = 3;

Вопрос 15:

Имеется фрагмент описания классов:

```
public class Class1
{
```

```
  public Class1() {}
  public Class1(int n) {}
  public Class1(double d) {}
}
public class Class2 extends Class1
{
  public Class2(String s) {}
}
```

и фрагмент описания и создания экземпляра класса:

```
Class2 cl = new <...>;
```

Вызов какого конструктора допустим в месте, помеченном <...>?

Метод ответа – выбор одного или нескольких ответов.

- Class2("Пример")
- Class2()
- Class2(5)
- Class2(5.2)
- Class2(1,2)

Вопрос 16:

Имеется фрагмент описания классов:

```
public class Class1
{
  public void m() {}
}
public class Class2 extends Class1
{
  public void m()
  {
    <...>
  }
}
```

Какой код, написанный вместо <...> позволит обратиться к методу m() класса Class1?

Метод ответа – ввод ответа.

Вопрос 17:

Имеется фрагмент описания классов:

```
public class Class1
{
  public void M() {}
```

```
}
public class Class2 extends Class1
{
  public <...> {}
}
```

Какой код, подставленный вместо <...> позволит заменить в потомке метод родителя M()?

Примечание: не ставить более одного пробела подряд.

Метод ответа – ввод ответа.

Вопрос 18:

Имеется фрагмент описания классов:

```
public abstract class Class1
{
  public int a;
  public abstract void b();
}
public class Class2 extends Class1
{
  public void c()
  {
    a = 3;
  }
}
```

Имеется ли синтаксическая ошибка во фрагменте?

Метод ответа – выбор только одного ответа.

– да, т.к. в классе Class2 не переопределен абстрактный метод класса-родителя

– да, недопустимо обращение к полям абстрактного класса из классов-потомков

– нет

– да, т.к. в классе-родителе нет метода c()

– да, т.к. при описании метода в классе Class2 не используется модификатор override

Раздел 3 Интерфейсы

Вопрос 1:

Имеется фрагмент программы:

```
public interface I1
{
  void a();
  int b();
}
```

Имеется ли ошибка в данном фрагменте?

Метод ответа – выбор только одного ответа.

– нет

– да, т.к. у методов не указаны модификаторы доступа

– да, т.к. у методов нет реализации

– да, т.к. в интерфейсе не может быть несколько методов

Вопрос 2:

Имеется фрагмент программы:

```
public interface I1
{
  void m();
}
class Class1 : I1
{
  public void m() {}
}
```

Имеется ли ошибка в данном фрагменте?

Метод ответа – выбор только одного ответа.

– нет

– да, т.к. реализация члена интерфейса не может иметь спецификатор доступа public

– да, т.к. описание наследования от интерфейса имеет вид: class Class1(I1)

– да, т.к. от интерфейса может наследовать не класс, а другой интерфейс

– да, т.к. у метода в интерфейсе не указана реализация

Вопрос 3:

Имеется фрагмент программы:

```
public interface I1
{
  void m(int m);
}
class Class1 : I1
{
  private int n;
  <...> m(int m) {n = m;}
}
```

Какой код, подставленный вместо <...> правильно реализует метод m?

Примечание: при вводе не ставить более одного пробела подряд.

Метод ответа – ввод ответа.

Вопрос 4:

```
public interface I1
{
  void m1(int m);
}
public interface I2 <...> I1
{
  void m2(int m);
}
```

Какое слово, подставленный вместо <...> указывает, что интерфейс I2 наследует от интерфейса I1?

Метод ответа – ввод ответа.

4 Выполнение домашних заданий

Задания для домашней работы, теоретический и методический материал для самостоятельного изучения представлены в методических указаниях.