

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Романчук Иван Сергеевич
Должность: Ректор
Дата подписания: 14.02.2025 09:52:52
Уникальный программный ключ:
6319edc2b582ffdacea443f01d5779368d0957ac34f5cd074d81181530452479

Приложение к рабочей программе дисциплины

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Наименование дисциплины	Открытые технологии разработки программного обеспечения
Направление подготовки / Специальность	02.03.03 Математическое обеспечение и администрирование информационных систем
Направленность (профиль) / Специализация	Технологии программирования и анализа больших данных
Форма обучения	очная
Разработчик(и)	Аврискин М.В., старший преподаватель кафедры программного обеспечения Павлова Е.А., старший преподаватель кафедры программного обеспечения

1. Темы дисциплины для самостоятельного освоения обучающимися

Стандарты языка Java, JDK, JRE.

Оновы синтаксиса Java: переменные, типы данных, операторы

Управляющие конструкции (условия, циклы)

Функции и работа с методами

Массивы и коллекции (Array, ArrayList, HashMap)

Объектно-ориентированное программирование (классы, объекты, наследование, инкапсуляция, полиморфизм)

Работа со строками

Исключения и обработка ошибок

Введение в многопоточное программирование

Основы работы с файлами (File, BufferedReader, BufferedWriter)

Основы программирования на языке Kotlin, преимущества по сравнению с Java

Фреймворки в экосистеме Java: JMIX, Spring и др.

2. План самостоятельной работы

№ п/п	Учебные встречи	Виды самостоятельной работы	Форма отчетности / контроля	Количество баллов	Рекомендуемый бюджет времени на выполнение (ак.ч.)*
1	2	3	4	5	6
1	Практические занятия	Изучить материалы лекций, обязательную и дополнительную литературу. Выполнение индивидуальных заданий. Подготовка к формам текущего контроля успеваемости.	Опрос на учебной встрече, программный код	Согласно баллам, установленным в МУП за работу на соответствующей встрече	5 часов на одну встречу
1	Аттестация	Подготовка к промежуточной аттестации по дисциплине	Контроль успеваемости согласно контрольным вопросам к зачету	Согласно баллам, установленным в РПД/МУП за работу на соответствующей встрече	10 часов
					Итого: 100

3. Требования и рекомендации по выполнению самостоятельных работ обучающихся, критерии оценивания

В течение семестра студенты применяют теоретические и практические знания, навыки и компетенции в области программирования Java-приложений.

Перед практическим занятием необходимо проработать содержание лекционных занятий и результатов выполнения практических заданий на предыдущих занятиях, рекомендуется также проработать материалы из списка литературы в МУП дисциплины.

На практических занятиях студентам требуется предоставить результаты самостоятельной работы:

- Программный код, решающий поставленную задачу.
- Ответы на вопросы преподавателя по теме учебной встречи.

При выдаче заданий на самостоятельную работу используется дифференцированный подход к студентам. Перед выполнением студентами самостоятельной внеаудиторной работы преподаватель проводит инструктаж по выполнению задания, который включает: цель задания, его содержание, сроки выполнения, ориентировочный объем работы, основные требования к результатам работы, критерии оценки. В процессе инструктажа преподаватель предупреждает студентов о возможных типичных ошибках, встречающихся при выполнении задания. Инструктаж проводится преподавателем за счет объема времени, отведенного на изучение дисциплины.

Система оценивания:

При проведении текущего контроля для оценки заданий применяется система оценивания:

- 100% от установленного балла - обучающийся своевременно предоставил корректно работающий программный код, продемонстрировал грамотное и логически правильное изложение ответа на вопросы, продемонстрировал способности самостоятельно и творчески решать сложные проблемы, навыки критического мышления.
- Оценка снижается:
 - При задержке представления корректного программного кода менее чем на 2 недели — на 10%;
 - При задержке представления корректного программного кода более чем на 2 недели — на 20%;
 - При неполных ответах на вопросы по коду – на 20–30%;

4. Рекомендации по самоподготовке к промежуточной аттестации по дисциплине

Обучающиеся, не набравшие 61 балла в течение семестра или не согласные с оценкой, полученной по итогам текущего контроля в семестре, проходят промежуточную аттестацию в форме экзамена. При проведении промежуточной аттестации результаты, полученные обучающимися в семестре, переводятся в формат традиционной оценки в соответствии со шкалой перевода баллов:

Ниже 61 балла – «неудовлетворительно»,

61-75 баллов – «удовлетворительно»,

76-90 баллов – «хорошо»,

91-100 баллов – «отлично».

Вопросы для промежуточной аттестации (ориентировочные формулировки вопросов)

1. Особенности языка программирования Java.
2. Основные понятия языка программирования высокого уровня: алфавит, лексемы.
Примеры.
3. Понятие типа данных, понятие переменной, понятие оператора в языке программирования высокого уровня. Примеры. Структура программы на языке высокого уровня. Примеры.
4. Система типов языка. Целые типы. Диапазон возможных значений, требуемая память, основные операции. Примеры.
5. Система типов языка высокого уровня. Вещественные типы. Диапазон возможных значений, требуемая память, основные операции. Примеры.
6. Система типов языка высокого уровня. Символьный тип. Диапазон возможных значений, требуемая память, основные операции. Примеры.
7. Система типов языка высокого уровня. Логический тип. Возможные значения, требуемая память, Основные отношения, операции. Примеры.

8. Понятие операции и выражения. Преобразования и совместимость типов в выражениях. Примеры.
9. Простые операторы языка высокого уровня. Оператор присваивания. Понятие совместимости типов (по присваиванию). Процедуры ввода и вывода. Примеры.
10. Операторы языка высокого уровня. Оператор перехода. Операторы прерывания. Примеры.
11. Условный оператор. Примеры.
12. Оператор выбора. Примеры.
13. Операторы цикла. Примеры.
14. Одномерные массивы. Описания массива. Возможные типы индексов, возможные типы элементов массива. Допустимые операции с массивами. Примеры.
15. Многомерные массивы. Описания массива. Возможные типы индексов, возможные типы элементов массива. Допустимые операции с массивами. Примеры.
16. Строковый тип. Хранение строк в памяти компьютера. Функции для работы со строками. Примеры.
17. Конструирование классов.
18. Инкапсуляция, особенности методов и конструкторов.
19. Наследование и полиморфизм. Абстрактные классы.
20. Особенности работы с классами.
21. Интерфейсы, внутренние классы.
22. Обобщения. Обобщённые методы, интерфейсы, конструкторы.
23. Язык программирования kotlin. Основы синтаксиса.
24. Язык программирования kotlin. Классы, поля, наследование.
25. Java Persistence API, сущности, связь со схемой базы данных.
26. Spring Framework.

Экзамен проводится в форме устного собеседования и решения практических задач-кейсов. Каждый билет содержит по 3 вопроса, из разных разделов курса. Преподаватель вправе задать уточняющий вопрос по каждому из вопросов билета. Итоговая оценка выводится как средняя из оценок по вопросам билета.

Ответ на каждый из вопросов оценивается по следующей шкале:

2 («неудовлетворительно») - студент не ответил на вопрос либо содержание ответа на раскрывает сути вопроса.

3 («удовлетворительно») - студент отвечает по существу, но не демонстрирует целостного представления по вопросу, не может аргументировать свой ответ.

4 («хорошо») - студент отвечает по существу, демонстрирует целостное представление по вопросу; не может аргументировать свой ответ либо аргументация не обоснована.

5 («отлично») - студент дает полный, развернутый, аргументированный ответ на вопрос.