

Комплект оценочных материалов по производственной практике
ПМ. 01 Разработка модулей программного обеспечения для компьютерных
систем по специальности
09.02.07. Информационные системы и программирование

Задания закрытого типа

Задания закрытого типа на выбор правильного ответа

Задание 1: Многопоточность в Java

Какой метод используется для запуска потока в Java после его создания?

- a) start()
- b) run()
- c) execute()

Правильный ответ: a)

Компетенции (индикаторы): ОК 5

Задание 2: Синхронизация потоков

Что делает ключевое слово synchronized в методе класса в Java?

- a) Запрещает создание новых потоков
- b) Обеспечивает взаимное исключение при доступе к ресурсу несколькими потоками
- c) Автоматически освобождает память от объекта

Правильный ответ: b)

Компетенции (индикаторы): ОК 4

Задание 3: Работа с файлами

Какой класс из пакета java.io используется для чтения текстового файла в Java?

- a) FileWriter
- b) BufferedReader
- c) FileInputStream

Правильный ответ: b)

Компетенции (индикаторы): ОК 3

Задание 4: Сборка мусора (Garbage Collection)

Что происходит, когда JVM вызывает метод System.gc() в Java?

- a) Немедленно освобождает всю память
- b) Запрашивает сборку мусора, но не гарантирует её выполнение
- c) Завершает все потоки программы

Правильный ответ: b)

Компетенции (индикаторы): ОК 2

Задание 5: Сетевое программирование

Какой класс используется для создания клиентского сокета в Java для TCP-соединения?

- a) DatagramSocket
- b) ServerSocket
- c) Socket

Правильный ответ: c)

Компетенции (индикаторы): ОК 1

Задания закрытого типа на установление соответствия

Задание 1: Установление соответствия (Многопоточность)

Какому методу соответствует описание: "Запускает новый поток выполнения в Java"?

- a) run()
- b) start()
- c) join()

Правильный ответ: b)

Компетенции (индикаторы): ОК 6

Задание 2: Установление соответствия (Синхронизация потоков)

Какому ключевому слову соответствует описание: "Обеспечивает взаимное исключение для предотвращения состояния гонки в многопоточных программах"?

- a) volatile
- b) synchronized
- c) transient

Правильный ответ: b)

Компетенции (индикаторы): ОК 7

Задание 3: Установление соответствия (Работа с файлами)

Какому классу соответствует описание: "Используется для записи текстовых данных в файл с буферизацией в Java"?

- a) FileReader
- b) BufferedWriter
- c) FileOutputStream

Правильный ответ: b)

Компетенции (индикаторы): ОК 8

Задание 4: Установление соответствия (Сборка мусора)

Какому механизму соответствует описание: "Автоматический процесс освобождения памяти от неиспользуемых объектов в JVM"?

- a) Just-In-Time компиляция
- b) Garbage Collection
- c) Bytecode верификация

Правильный ответ: b)

Компетенции (индикаторы): ОК 9

Задание 5: Установление соответствия (Сетевое программирование)

Какому классу соответствует описание: "Создает серверный сокет, ожидающий входящих TCP-соединений в Java"?

- a) Socket
- b) DatagramSocket
- c) ServerSocket

Правильный ответ: c)

Компетенции (индикаторы): ПК 1.3,1.2

Задания закрытого типа на установление правильной последовательности

Задание 1: Установление правильной последовательности (Многопоточность)

Правильная последовательность создания и запуска потока в Java:

1. Создать объект класса, реализующего Runnable или наследующего Thread.
 2. Переопределить метод run() для выполнения кода в потоке.
 3. Вызвать метод start() для запуска потока.
- a) 1, 2, 3
b) 2, 1, 3
c) 1, 3, 2

Правильный ответ: а)

Компетенции (индикаторы): ПК 1.3,1.2

Задание 2: Установление правильной последовательности (Синхронизация потоков)

Правильная последовательность использования synchronized блока для защиты критической секции в Java:

1. Определить объект-монитор.
 2. Заключить критический код в synchronized блок с этим объектом.
 3. Потоки автоматически получают доступ по очереди.
- a) 1, 2, 3
b) 2, 1, 3
c) 3, 1, 2

Правильный ответ: а)

Компетенции (индикаторы): ПК 1.3,1.2

Задание 3: Установление правильной последовательности (Работа с файлами)

Правильная последовательность чтения данных из текстового файла с использованием BufferedReader в Java:

1. Создать объект FileReader с путем к файлу.
 2. Обернуть его в BufferedReader для буферизации.
 3. Вызвать методы readLine() или read() для чтения данных.
- a) 1, 2, 3
b) 2, 1, 3
c) 3, 1, 2

Правильный ответ: а)

Компетенции (индикаторы): ОК 6

Задание 4: Установление правильной последовательности (Сборка мусора)

Правильная последовательность этапов работы сборщика мусора в JVM (упрощенная модель):

1. Пометка неиспользуемых объектов.
 2. Освобождение памяти от помеченных объектов.
 3. Компактизация оставшейся памяти (опционально).
- a) 1, 2, 3
b) 2, 1, 3
c) 1, 3, 2

Правильный ответ: а)

Компетенции (индикаторы): ОК 7

Задание 5: Установление правильной последовательности (Сетевое программирование)

Правильная последовательность создания TCP-сервера с использованием ServerSocket в Java:

- Создать объект ServerSocket на определенном порту.
Вызвать accept() для ожидания входящего соединения.
Получить объект Socket для общения с клиентом.

- a) 1, 2, 3
b) 2, 1,

Компетенции (индикаторы): ОК 6

Задания открытого типа

Задания открытого типа на дополнение

Напишите пропущенное слово (словосочетание).

1. Процесс, посредством которого проверяется правильность программы называется _____.

Правильный ответ: тестирование

Компетенции: ОК 1, ОК 2, ПК 1.4

2. _____ — совокупность действий по обеспечению работоспособности и актуальности программного обеспечения.

Правильный ответ: поддержка

Компетенции: ОК 2, ПК 1.4.

3. _____ — процесс поиска дефектов в программе.

Правильный ответ: отладка

Компетенции: ОК 2, ПК 1.3.

4. _____ — конкретный набор условий, входных данных и ожидаемых результатов, используемый для проверки определенной функциональности программы (ПО).

Правильный ответ: тест-кейс

Компетенции: ОК 2, ПК 1.4.

5. _____ — документ, описывающий все этапы тестирования, цели и задачи, а также критерии оценки для конкретного проекта.

Правильный ответ: План тестирования

Компетенции: ОК 10, ПК 1.4.

6. _____ — метод отладки, при котором программа выполняется шаг за шагом для отслеживания значений переменных.

Правильный ответ: Трассировка

Компетенции: ОК 9, ПК 1.3.

7. Процесс улучшения структуры существующего кода без изменения его внешнего поведения называется _____.

Правильный ответ: рефакторинг

Компетенции: ПК 1.5

8. Метка в коде, которая приостанавливает выполнение программы в определенной точке для анализа состояния переменных и потока выполнения, называется _____.

Правильный ответ: точка останова (breakpoint)

Компетенции: ПК 1.3

9. Метод рефакторинга, при котором часть кода выделяется в отдельный метод с понятным названием, называется _____.

Правильный ответ: извлечение метода (extract method)

Компетенции: ПК 1.5

10. Процесс изменения кода с целью повышения скорости выполнения или уменьшения потребления ресурсов называется _____.

Правильный ответ: оптимизация

Компетенции: ПК 1.5

Задания открытого типа с кратким свободным ответом

1. Какой документ описывает функциональные требования к программному обеспечению?

Правильный ответ: Техническое задание

Компетенции: ОК 10, ПК 1.4.

2. Какой из типов тестирования фокусируется на производительности приложения?

Правильный ответ: Нагрузочное тестирование

Компетенции: ОК 2, ПК 1.4.

3. Что такое валидация (validation) в контексте тестирования программного обеспечения?

Правильный ответ: подтверждение соответствия требованиям пользователя

Компетенции: ОК 2, ПК 1.4.

4. Что такое проверка (verification) в контексте тестирования программного обеспечения?

Правильный ответ: подтверждение соответствия спецификациям

Компетенции: ОК 2, ПК 1.4.

5. Как называется вид тестирования, направленный на выявление уязвимостей и обеспечение защиты данных?

Правильный ответ: тестирование безопасности

Компетенции: ОК 2, ПК 1.4.

6. Как называется тестирование, проводимое конечными пользователями для оценки готовности продукта к выпуску?

Правильный ответ: приёмочное тестирование (или пользовательское тестирование)

Компетенции: ОК 2, ПК 1.4.

7. Как называется инструмент, который позволяет выполнять программу пошагово и отслеживать значения переменных?

Правильный ответ: отладчик (debugger)

Компетенции: ПК 1.3

8. Какой метод отладки предполагает вывод промежуточных значений переменных в консоль или файл?

Правильный ответ: логгирование (logging)

Компетенции: ПК 1.3

9. Какой прием рефакторинга следует применить, если в коде многократно повторяется один и тот же фрагмент?

Правильный ответ: вынести повторяющийся код в отдельный метод

Компетенции: ПК 1.5

10. Что такое "запахи кода" (code smells)?

Правильный ответ: признаки потенциальных проблем в коде, указывающие на необходимость рефакторинга

Компетенции: ПК 1.5

11. Какой инструмент используется для измерения производительности кода и выявления "узких мест"?

Правильный ответ: профайлер (profiler)

Компетенции: ПК 1.5

Задания открытого типа с развернутым ответом

1. Отладка критической ошибки в модуле работы с массивами

Задачи:

- найти причину ошибки сегментации в функции обработки массива
- использовать отладчик GDB для анализа падения программы
- предложить исправление кода

Исходный код для анализа:

```
#include <iostream>
void processArray(int *arr, int size) {
    for (int i = 0; i <= size; ++i)
    {    arr[i] *= 2;    } // Ошибка здесь
}
int main() {
    int size = 5;
    int *array = new int[size];
    // Инициализация массива
    for (int i = 0; i < size; ++i) {
        array[i] = i + 1;
```

```
    }
    processArray(array, size);
    // Вывод результата (проверка)
    for (int i = 0; i < size; ++i) {
        std::cout << array[i] << " ";
    }
    std::cout << std::endl;
    delete[] array;
    array = nullptr;
    return 0;
}
```

Время выполнения – 5 мин.

Ожидаемый ответ (один из возможных вариантов):

Диагностика в GDB: Запуск: gdb ./program; Установка точки останова: break process_array; Запуск: run; Анализ падения: backtrace

Найденная ошибка:

Выход за границы массива в условии цикла $i \leq size$ вместо $i < size$

Исправление кода:

```
void processArray(int *arr, int size) {
    for (int i = 0; i < size; ++i) // Ошибка здесь
    {
        arr[i] *= 2;
    }
}
```

Критерии оценивания:

- правильное использование команд GDB для диагностики
- точное определение причины ошибки
- корректное исправление кода
- понимание работы с динамической памятью

Компетенции: ОК 1, ОК 2, ПК 1.3

2. Отладка бесконечного цикла в модуле математических вычислений

Задачи:

- использовать отладчик для анализа бесконечного цикла
- найти логическую ошибку в условии выхода из цикла
- предложить корректный алгоритм

Исходный код для анализа:

```
#include <iostream>
int calFactorial(int n) {
    int result = 1;
    int i = 1;

    while (i <= n) { // Ошибка в условии
        result *= i;
    }
    return result;
}
int main() {
```

```
int n = 5; // Число, для которого вычисляем факториал
int fact = calcFactorial(n);
std::cout << "Факториал " << n << " равен: " << fact << std::endl;
return 0; }
```

Время выполнения – 5 мин.

Ожидаемый ответ (один из возможных вариантов):

Диагностика в отладчике: Запуск программы в отладчике; Установка точки останова в цикле; Наблюдение за изменением переменной *i*; Обнаружение, что *i* не изменяется.

Найденная ошибка:

Отсутствует инкремент переменной *i* в цикле

Исправление кода:

```
while (i <= n) { result *= i; i++; } // Добавить инкремент
```

Критерии оценивания:

- умение использовать отладчик для анализа циклов
- правильное определение причины бесконечного цикла
- точное исправление логической ошибки
- понимание работы циклов и условий

Компетенции: ОК 1, ОК 2, ПК 1.3

Ты придумал три задания открытого типа с развернутым ответом для компетенции ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств по дисциплине МДК.01.02 Поддержка и тестирование программных модулей В ответе укажите задачи задания, Время выполнения, Ожидаемый ответ, Критерии оценивания (не оценка, а критерий) Мне не нравятся Можешь более конкретные задания с конкретным ответом , а то это какие-то размытые задания и ответы

Задания для компетенции ПК 1.4 (Тестирование программных модулей)

Задание 1: Разработка стратегии тестирования сложного модуля

Ситуация: Вам необходимо протестировать модуль расчета страховых премий, который учитывает множество факторов: возраст клиента, историю вождения, тип автомобиля, регион проживания. Модуль содержит сложную логику вычислений с множеством условий и исключений.

Вопрос: Разработайте комплексную стратегию тестирования данного модуля. В ответе отразите:

1. Какие виды тестирования вы примените и почему
2. Как будете разрабатывать тестовые случаи (приведите примеры 3-4 ключевых тест-кейсов)
3. Какие тестовые данные подготовите для покрытия различных сценариев
4. Как будете проверять граничные условия и исключительные ситуации

Критерии оценивания:

- Полнота стратегии тестирования (3 балла)
- Релевантность тест-кейсов и тестовых данных (3 балла)
- Учет различных видов тестирования (2 балла)
- Проработанность проверки граничных условий (2 балла)

Компетенции: ПК 1.4

Задание 2: Тестирование модуля с внешними зависимостями

Ситуация: Модуль обработки платежей взаимодействует с внешними системами: банковским шлюзом, системой учета и службой нотификаций. Необходимо обеспечить надежное тестирование модуля.

Вопрос: Опишите ваш подход к тестированию модуля с внешними зависимостями.

Включите:

1. Как будете изолировать модуль для юнит-тестирования
2. Какие mock-объекты или заглушки создадите
3. Как организуете интеграционное тестирование
4. Как будете тестировать обработку ошибок при работе с внешними системами

Критерии оценивания:

- Понимание изоляции зависимостей (3 балла)
- Качество описания mock-объектов (3 балла)

- Логичность подхода к интеграционному тестированию (2 балла)
- Проработанность тестирования ошибок (2 балла)

Компетенции: ПК 1.4

Задание 3: Регрессионное тестирование после доработок

Ситуация: В существующий модуль отчетности добавили новую функциональность – экспорт в дополнительный формат. Модуль содержит 150+ тестов, которые выполняются 40 минут.

Вопрос: Разработайте стратегию регрессионного тестирования. Опишите:

1. Как будете отбирать тесты для регрессии
2. Какие методы ускорения тестирования примените
3. Как организуете проверку новой функциональности
4. Как обеспечите сохранение существующей функциональности

Критерии оценивания:

- Эффективность отбора тестов (3 балла)
- Знание методов оптимизации тестирования (3 балла)
- Полнота проверки новой функциональности (2 балла)
- Обеспечение сохранности существующей функциональности (2 балла)

Компетенции: ПК 1.4

Задания для компетенции ПК 1.5 (Рефакторинг и оптимизация)

Задание 1: Рефакторинг legacy-кода

Ситуация: Вам передан модуль учета сотрудников с кодом, написанным 5 лет назад. Код содержит: методы длиной 200+ строк, сложные вложенные условия, магические числа, повторяющиеся фрагменты.

Вопрос: Разработайте план рефакторинга. Опишите:

1. Какие "запахи кода" выявите в первую очередь
2. Какие приемы рефакторинга примените и в какой последовательности
3. Как будете обеспечивать безопасность изменений
4. Какие метрики качества будете отслеживать

Критерии оценивания:

- Глубина анализа проблем кода (3 балла)
- Обоснованность выбора приемов рефакторинга (3 балла)
- Проработанность стратегии безопасности изменений (2 балла)
- Релевантность метрик качества (2 балла)

Компетенции: ПК 1.5

Задание 2: Оптимизация модуля обработки данных

Ситуация: Модуль анализа логов работает с большими объемами данных (1+ ГБ ежедневно). Время обработки постоянно растет, клиенты жалуются на медленную работу.

Вопрос: Разработайте план оптимизации производительности. Включите:

1. Как будете выявлять узкие места (инструменты, методы)
2. Какие алгоритмические оптимизации предложите
3. Как будете работать с памятью и вводом-выводом
4. Как оцените эффективность оптимизаций

Критерии оценивания:

- Комплексность подхода к выявлению узких мест (3 балла)

- Эффективность предлагаемых оптимизаций (3 балла)
- Учет ресурсов памяти и I/O (2 балла)
- Качество оценки результатов (2 балла)

Компетенции: ПК 1.5

Задание 3: Рефакторинг архитектуры модуля

Ситуация: Модуль авторизации вырос из одного класса в сложную структуру с нарушением принципов SOLID. Высокая связанность затрудняет тестирование и доработки.

Вопрос: Опишите архитектурный рефакторинг модуля. Рассмотрите:

1. Какие архитектурные проблемы выявите
2. Как перепроектируете модуль с учетом принципов SOLID
3. Какие шаблоны проектирования примените
4. Как будете проводить рефакторинг без простоя системы

Критерии оценивания:

- Глубина анализа архитектурных проблем (3 балла)
- Корректность применения принципов SOLID (3 балла)
- Уместность использования шаблонов проектирования (2 балла)
- Практичность плана миграции (2 балла)

Компетенции: ПК 1.5

Обоснование: Задания разработаны для проверки практических навыков тестирования, рефакторинга и оптимизации, требуют глубокого понимания процессов обеспечения качества программного обеспечения и владения современными методами разработки, что полностью соответствует компетенциям ПК 1.4 и ПК 1.5. Придумай три задания открытого типа с развернутым ответом для компетенции ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств по дисциплине МДК.01.02 Поддержка и тестирование программных модулей

В ответе укажите задачи задания, Время выполнения, Ожидаемый ответ, Критерии оценивания (не оценка, а критерий)