Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»
Кафедра вычислительной техники

Паспорт зачета

по дисциплине «Инжиниринг программного обеспечения», 5 семестр

1. Методика оценки

Зачет проводится в устной (письменной) форме, по билетам. Билет состоит из 3 вопросов и формируется по следующему правилу:

- первый вопрос выбирается из диапазона вопросов 1-10;
- второй вопрос из диапазона вопросов 11-20;
- третий вопрос – задача.

Таким образом, проверяются результаты обучения по модулю, соотнесенные с установленными в программе индикаторами достижения компетенций.

На зачете преподаватель вправе задавать студенту уточняющие и дополнительные вопросы из общего перечня (п. 4).

Форма билета для зачета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Факультет АВТФ

Билет № _____
к зачету по дисциплине «Инжиниринг программного обеспечения»

1. Вопрос 1.
2. Вопрос 2.
3. Задача.

Утверждаю: зав. кафедрой ____ _____ должность, ФИО
(подпись)
(дата)

2. Критерии оценки результатов обучения, соотнесенных с уровнями освоения индикаторов достижения компетенций

Ответ на билет для зачета засчитывается на **продвинутом** уровне, если студент проводит сравнительный комплексный анализ материала, выявляет проблемы, предлагает механизмы их решения, представляет количественные характеристики определенных процессов, приводит конкретные примеры, не допускает ошибок и способен обосновать выбор метода решения задачи. Совокупность результатов обучения по дисциплине

(модулю) и соотнесенных с ними индикаторов достижения компетенций не содержит пробелов. Установленные в программе компетенции сформированы в полном объеме. Оценка составляет *от 35 до 40 баллов.*

Ответ на билет для зачета засчитывается на **базовом** уровне, если студент при ответе на вопросы формулирует основные понятия, дает характеристику процессов, явлений, проводит анализ причин, условий, способен представить количественные и качественные характеристики процессов, не допускает существенных ошибок при решении задачи. Совокупность результатов обучения по дисциплине (модулю) и соотнесенных с ними индикаторов достижения компетенций содержит несущественные пробелы. Установленные в программе компетенции сформированы на базовом уровне. Оценка составляет *от 34 до 27 баллов.*

Ответ на билет для зачета засчитывается на **пороговом** уровне, если студент при ответе на вопросы дает определение основных понятий, может показать причинно-следственные связи явлений, при решении задачи допускает непринципиальные ошибки, например, вычислительные. Совокупность результатов обучения по дисциплине (модулю) и соотнесенных с ними индикаторов достижения компетенций содержит пробелы. Установленные в программе компетенции сформированы на пороговом уровне. Оценка составляет *от 26 до 20 баллов.*

Ответ на билет для зачета считается **неудовлетворительным** (ниже порогового уровня), если студент при ответе на вопросы не дает определений основных понятий, не способен показать причинно-следственные связи явлений, при решении задачи допускает принципиальные ошибки. Совокупность результатов обучения по дисциплине (модулю) и соотнесенных с ними индикаторов достижения компетенций содержит существенные пробелы. Установленные в программе компетенции не сформированы. Оценка составляет *менее 20 баллов.*

3. Шкала оценки

Зачет считается сданным, если сумма баллов по всем заданиям билета составляет от 10 до 20 баллов включительно. Сумма менее 10 баллов признается неудовлетворительным результатом промежуточной аттестации по модулю.

В общей оценке по дисциплине баллы за зачет учитываются в соответствии с правилами балльно-рейтинговой системы, установленными в НГТУ.

4. **Вопросы к** зачету **по дисциплине «**Инжиниринг программного обеспечения**»**

1. Классы языка Java и их синтаксис. Члены класса. Статические члены класса. Поля и методы класса. Главный метод приложения. Конструкторы класса. Цепочки конструкторов. Блоки инициализации. Статические поля и методы классов. Инициализация статических полей класса.
2. Пакеты классов и интерфейсов. Импорт классов и интерфейсов из других пакетов. Разновидности видимости классов и интерфейсов. Соглашения об именовании пакетов, классов, директорий и файлов при программировании на Java.
3. Наследование полей и методов классов в языке Java. Перекрытие наследуемых методов. Использование конструкторов наследуемых классов. Разновидности видимости членов классов. Полиморфизм. Абстрактные классы. Конечные (final) классы.
4. Вложенность классов. Статические вложенные и внутренние классы. Доступ к статическим вложенным классам. Внутренние классы. Доступ к внутренним классам. Внутренние локальные классы. Внутренние анонимные классы.
5. Интерфейсы языка Java. Типизация и интерфейсы. Члены интерфейсов языка Java. Наследование интерфейсов. Реализация интерфейсов классами. Назначение и

использование интерфейсов при программировании на языке Java.
6. Типы-перечисления. Поля и методы типов-перечислений. Аннотации. Объявление типа аннотации и его использование. Аннотации для компилятора.
7. Исключения (exceptions) в языке Java. Порождение и перехват исключений. Список исключений в объявлении метода. Источники исключений. Иерархия исключений. Приоритеты при обработке исключений. Уведомления (assert) в языке Java. Использование исключений и уведомлений при программировании на Java.
8. Стандартная библиотека коллекций языка Java. Интерфейсы, реализации и алгоритмы коллекций. Структура библиотеки коллекций. Коллекции, множества и списки. Использование реализаций интерфейсов коллекций. Карты (maps) в библиотеке коллекций. Использование различных реализаций карт. Итераторы карт и коллекций. Стандартные алгоритмы библиотеки для работы с коллекциями и массивами.
9. Методы рефакторинга для преобразования структуры программы на языке Java. Рефакторинг типов. Рефакторинг иерархии наследования. Перемещение методов по иерархии наследования и между классами. Использование среды Eclipse для рефакторинга программы на языке Java.
10. Родовые типы в языке Java. Назначение родовых типов. Не ковариантность родовых типов. Родовой тип wildcard. Родовые методы. Ограниченные родовые типы.
11. Потоки байтового вывода языка Java. Назначение и возможности классов OutputStream, ByteArrayOutputStream, FileOutputStream, PipedOutputStream, FilterOutputStream, BufferedOutputStream, DataOutputStream, PrintStream. Потоки символьного вывода языка Java.
12. Потоки ввода языка Java. Назначение и возможности классов InputStream, ByteArrayInputStream, FileInputStream, PipedInputStream, FilterInputStream, BufferedInputStream, DataInputStream. Потоки символьного ввода языка Java. Чтение данных из потока с помощью класса Scanner.
13. Java DataBase Connectivity (JDBC) - стандарт взаимодействия Java-приложений с различными СУБД. Подключение драйверов СУБД. Выполнение SQL команд. Управление транзакциями. Использование JDBC для извлечения информации о структуре (метаданных) базы данных. Визуализация структуры базы данных.
14. Интернационализация программ в языке Java. Файлы текстовых ресурсов. Выбор языка пользователя для выдачи текстовых сообщений.
15. Java DataBase Connectivity (JDBC) - стандарт взаимодействия Java-приложений с различными СУБД. Подключение драйверов СУБД. Выполнение SQL команд. Управление транзакциями. Использование JDBC для извлечения информации о структуре (метаданных) базы
16. Лямбда выражения в языке Java8. Замена анонимных классов лямбда выражениями. Функциональные интерфейсы. Синтаксис Lambda-выражений. Примеры функциональных интерфейсов из пакета java.util.function.
17. Потоки в языке Java. Определение потоков. Отличие операций потоков от операций коллекций. Обработка данных потока в конвейере. Методы для порождения потоков. Методы для преобразования потоков в конвейере. Преобразования в конце конвейера.