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

Паспорт экзамена

по дисциплине «Инжиниринг программного обеспечения», 6 семестр

1. Методика оценки

Экзамен проводится в устной (письменной) форме, по билетам. Билет состоит из N вопросов и задачи 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 дисциплине, соотнесенные с установленными в программе индикаторами достижения компетенций.

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

Форма экзаменационного билета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Факультет АВТФ

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

1. Вопрос 1
2. Вопрос 2
3. Вопрос 3

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

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

Ответ на экзаменационный билет 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 (если задача предусмотрена). 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 экзаменационный билет (тест) 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 (если задача предусмотрена). 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 экзаменационный билет засчитывается на **пороговом** уровне, если студент при ответе на вопросы дает определение основных понятий, может показать причинно-следственные связи явлений, при решении задачи (если задача предусмотрена) 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 экзаменационный билет 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 (если задача предусмотрена) 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

Экзамен считается сданным, если сумма баллов по всем заданиям билета составляет от 20 до 40 баллов включительно. Сумма менее 20 баллов признается неудовлетворительным результатом промежуточной аттестации по дисциплине.

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

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

1. Процессы, потоки и приоритеты в приложении. Реализация многозадачности.
2. Выражения и операции. Приведение типов. Класс Object. Упаковка и распаковка примитивных типов. Операторы языка Java.
3. Массивы в Java. Создание и инициализация. Передача в функции. Методы обработки массивов.
4. Объявление класса. Компоненты класса. Создание объектов класса. Модификаторы доступа к классам и членам классов.
5. Методы класса. Лямбда-выражения.
6. Интерфейсы. Реализация интерфейсов. Методы интерфейсов по умолчанию. Множественное наследование интерфейсов.
7. Вложенные и внутренние, локальные классы. Создание объектов этих классов.
8. Наследование. Реализация наследования. Переопределение методов родительского класса. Полиморфизм. Абстрактные классы.
9. Обработка исключений. Проверяемые и непроверяемые исключения. Создание

пользовательских классов исключений.
10. Описание классов ArrayList, LinkedList, HashSet, TreeSet, HashMap, TreeMap, в том числе структуру данных.
11. Поточная модель Java. Базовые классы создания потоков. Состояние потоков. Приоритеты. Группы потоков. Потоки-демоны.
12. Синхронизация потоков. Монитор. Управление монитором. Передача данных между потоками.
13. Параллельное API. Пулы потоков.
14. Строки в Java. Обработка строк. Регулярные выражения.
15. Классы-коллекций в Java. Иерархии классов коллекций. Принципы их внутренней организации.
16. Потоки ввода/вывода. Классификации потоков. Создание потоков.
17. Классы-фильтры. Применение классов фильтров. Буферизация.
18. Сериализация объектов. Управляемая пользователем сериализация. Десериализация.
19. События. Интерфейсы прослушивания событий. Модель делегирования событий.
20. События. Классы-адаптеры. Использование вложенных и анонимных внутренних классов для обработки событий.
21. Графические библиотеки Java. Принципы разработки графического интерфейса. Контейнеры, менеджеры компоновки, меню, диалоговые окна. Модальные и немодальные окна.
22. Работа в сети. Протокол TCP/IP. IP-адрес хоста. Класс InetAddress. URL. Работа с ресурсами, заданными адресами URL.
23. Создание клиент-серверных приложений. Сокеты. Передача данных по протоколу TCP.
24. JDBC. Соединение с базой данных. Выполнение SQL-запросов к БД и обработка результата.
25. Реализация классов и наследования в Java на примере классов котов и собак, наследование интерфейсов и реализация основных команд. Многопоточность