Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»
Кафедра автоматизированных систем управления
Кафедра автоматики
Кафедра вычислительной техники
Кафедра защиты информации

Паспорт экзамена

по дисциплине «Технологии и методы программирования», 4 семестр

Методика оценки

Экзамен проводится в устной форме по билетам. Билет состоит из 2 вопросов и формируется по следующему правилу:
первый вопрос выбирается из диапазона вопросов 1–20;
второй вопрос выбирается из диапазона вопросов 21–40.

Таким образом, проверяются результаты обучения по дисциплине, соотнесённые с установленными в программе индикаторами достижения компетенций ОПК-2.1 и ОПК-2.2.

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 «Технологии и методы программирования»

1. Вопрос 1
2. Вопрос 2.

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

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. Совокупность результатов обучения по дисциплине 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 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. Совокупность результатов обучения по дисциплине и соотнесенных с ними индикаторов достижения компетенций содержит несущественные пробелы. Установленные в программе компетенции сформированы на базовом уровне. Оценка составляет *от 27 до 34 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, допускает непринципиальные ошибки. Совокупность результатов обучения по дисциплине и соотнесенных с ними индикаторов достижения компетенций содержит пробелы. Установленные в программе компетенции сформированы на пороговом уровне. Оценка составляет *от 20 до 26 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, допускает принципиальные ошибки. Совокупность результатов обучения по дисциплине и соотнесенных с ними индикаторов достижения компетенций содержит существенные пробелы. Установленные в программе компетенции не сформированы. Оценка составляет *менее* 20 *баллов.*

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.

Вопросы к экзамену **по дисциплине «**Технологии и методы программирования**»**

1. Основные структуры данных и критерии их выбора. Влияние операций доступа, поиска и изменения на архитектуру программы.
2. Объектно-ориентированное программирование: инкапсуляция, наследование, полиморфизм и композиция.
3. Типизация в языках программирования: статическая и динамическая, строгая и нестрогая; последствия преобразования типов.
4. Принципы работы современных программных средств. Критерии выбора инструментов разработки, в том числе отечественного производства.
5. Клиент-серверное взаимодействие: роли frontend и backend, запрос и ответ, границы ответственности.

6. HTML: структура документа, семантические элементы, формы и организация пользовательского ввода.
7. CSS: каскад, наследование, селекторы и блочная модель.
8. Адаптивная вёрстка: Flexbox, Grid, медиазапросы и проверка интерфейса на разных размерах экрана.
9. JavaScript: типы данных, функции, области видимости и обработка ошибок.
10. DOM и события: изменение документа, обработка пользовательских действий и работа с формами.
11. Асинхронность JavaScript: Promise, async/await, сетевые запросы и обработка неуспешных ответов.
12. Проверка входных данных на клиенте и сервере. Безопасное отображение пользовательского содержимого.
13. Контроль версий: репозиторий, фиксация изменений, ветки, слияние и разрешение конфликтов.
14. Организация командной работы с системой контроля версий. Проверка изменений, история и воспроизводимость сборки.
15. Frontend-фреймворки: компоненты, состояние, передача данных и критерии выбора.
16. Системы управления базами данных. Реляционная модель, таблицы, ключи и связи.
17. Проектирование базы данных: сущности, атрибуты, кардинальности и переход от предметной области к схеме.
18. Нормализация отношений: функциональные зависимости, нормальные формы и устранение аномалий.
19. SQL: создание таблиц, типы данных и ограничения целостности.
20. SQL: выборка, фильтрация, сортировка, группировка и агрегатные функции.
21. Соединения таблиц и подзапросы. Выбор способа получения связанных данных.
22. Изменение данных в SQL: добавление, обновление и удаление. Проверка условий и сохранение целостности.
23. Транзакции, свойства ACID и согласованность данных при совместном доступе.
24. Индексы, планы выполнения и оценка производительности запросов.
25. Архитектура приложения: разделение на слои, зависимости, интерфейсы и модульность.
26. Архитектурные подходы и паттерны проектирования. Обоснование выбора паттерна для конкретной задачи.
27. Backend-приложение: маршрутизация, обработка запроса, бизнес-логика и доступ к данным.
28. API: контракт, формат данных, HTTP-методы, коды состояния и документирование.
29. Проектирование и проверка API: валидация, обработка ошибок, версии и совместимость клиентов.
30. Информационная безопасность приложения: разграничение доступа, защита от SQL-инъекций и безопасная работа с конфигурацией.
31. Тестирование программного обеспечения: цели, уровни, виды и критерии завершения.
32. Модульное, интеграционное и системное тестирование. Изоляция зависимостей и тестовые двойники.

33. Проектирование тестов: классы эквивалентности, граничные значения, позитивные и негативные сценарии.
34. Автоматизация тестирования интерфейса, базы данных и API. Фикстуры, повторяемость и оформление дефекта.
35. Регрессионное тестирование и организация проверок при изменении программного продукта.
36. Нейронная сеть: нейрон, веса, функция активации, архитектура и задача обучения.
37. Подготовка данных, обучающая и проверочная выборки, метрики и переобучение нейронных сетей.
38. Интеграция нейросетевого компонента в приложение: интерфейс, ограничения, проверка качества и безопасная обработка данных.
39. Методологии разработки: каскадная, итерационная и гибкая модели; критерии выбора.
40. Scrum и Kanban: организация работы, планирование, приоритеты, обратная связь и управление изменениями требований.