

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Порохня Андрей Александрович
Должность: И.о. директора института перспективной инженерии
Дата подписания: 19.06.2026 16:00:03
Уникальный программный ключ:
990bea3aeec48c23bd8699e48288f8d1960c600

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ**
**Федеральное государственное автономное образовательное учреждение
высшего образования**
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ

И.о. директора института
перспективной инженерии,
канд. техн. наук, доцент
А.А. Порохня

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

для проведения текущего контроля успеваемости и промежуточной аттестации по
дисциплине

Алгоритмы back-end разработки веб-приложений

Направление подготовки	<u>09.03.04 Программная инженерия</u>
Направленность (профиль)	<u>Разработка и сопровождение программного обеспечения</u>
Форма обучения	очная
Год начала обучения	2026
Реализуется в	4, 5 семестре

Введение

1. Назначение: Оценочные материалы (оценочные средства) предназначены для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся, и представляют собой совокупность контрольно-измерительных материалов и методов их использования для измерения уровня достижения обучающимися установленных результатов обучения.

2. ФОС является приложением к программе дисциплины «Алгоритмы back-end разработки веб-приложений».

3. Разработчик Новикова Е.Н., заведующая межинститутской базовой кафедры департамента цифровых, робототехнических систем и электроники

4. Проведена экспертиза ФОС.

Члены экспертной группы:

Председатель: Новикова Елена Николаевна, зав. межинститутской базовой кафедрой

(ФИО, должность)

Члены комиссии:

Николаев Евгений Иванович, доцент департамента цифровых и робототехнических систем и электроники.

(ФИО, должность)

Винокурский Д. Л., доцент департамента цифровых, робототехнических систем и электроники

(ФИО, должность)

Представитель организации-работодателя: А. В. Новиков, начальник отдела АСУТП ООО «ЕНДС-Ставрополь»

(ФИО, должность)

Экспертное заключение: Оценочные материалы (оценочные средства) по дисциплине «Алгоритмы back-end разработки веб-приложений» составлены в соответствии с требованиями самостоятельно устанавливаемого Федерального государственного образовательного стандарта высшего образования по направлению 09.03.04 «Программная инженерия» и позволяют оценить уровень сформированности компетенции ПК-2, ПК-3, ПК-8.

5. Срок действия ФОС определяется сроком реализации образовательной программы.

.

1. Описание показателей и критериев оценивания на различных этапах их формирования, описание шкал оценивания

Уровни сформированности компетенци(ий), индикатора (ов)	Дескрипторы			
	Минимальный уровень не достигнут (Неудовлетворительно) 2 балла	Минимальный уровень (удовлетворительно) 3 балла	Средний уровень (хорошо) 4 балла	Высокий уровень (отлично) 5 баллов
Компетенция: ПК-2 – Способен выполнять проектирование и разработку архитектуры программных систем среднего и крупного масштаба с применением современных паттернов и методов проектирования				
Результаты обучения по дисциплине (модулю): <i>Индикатор:</i> ПК-2 _{ид-1} – Выделяет и анализирует функциональные и нефункциональные требования (FURPS+), выбирает архитектурный стиль (монолит, микросервисы, EDA, Clean Architecture) на основе сценариев качества	Не знает алгоритмы консистентного хеширования, не может объяснить, зачем нужен распределённый кэш	Знает теорию consistent hashing, но не может реализовать; знает названия 2Q/ARC, но не отличает их	Реализует consistent hashing с виртуальными узлами на Java/Python/Go, настраивает базовый LRU-кэш, использует gossip для обнаружения отказов в кластере из 3 узлов	Реализует consistent hashing и rendezvous hashing, внедряет алгоритм TinyLFU или 2Q, настраивает Phi Accrual Failure Detector и полный gossip-протокол для динамической кластеризации
Компетенция: ПК-3 – Владеет навыками разработки программного обеспечения: способен реализовывать алгоритмы, писать код на современных языках программирования, создавать веб- и desktop-приложения, работать с базами данных				
Результаты обучения по дисциплине (модулю): <i>Индикатор:</i> ПК-3 _{ид-1} – Разрабатывает алгоритмы и реализует их на современных языках программирования (C++, C#, Python, Java, JavaScript/TypeScript, Go) с использованием эффективных структур данных и стандартных библиотек	Не может написать LRU даже через List, не знает про rate limiting	Реализует LRU через список и словарь (O(1) в среднем), но не LFU; реализует простой rate limiter на fixed window	Реализует LRU и LFU с O(1), реализует алгоритм взвешенного round-robin (SWRR) для балансировки, настраивает sliding window или token bucket	Реализует LFU с подсчётом частоты и фильтром Блума, реализует Power of Two Choices, использует GCRA (Generic Cell Rate Algorithm) для rate limiting с высокой точностью

Компетенция: ПК-8 – Способен адаптировать и применять методы искусственного интеллекта и машинного обучения для решения профессиональных задач, включая интеграцию готовых моделей в программные продукты

Результаты обучения по дисциплине (модулю): <i>Индикатор:</i> ПК-3_{ид-1} – Интегрирует предобученные модели и API ИИ (Hugging Face, OpenAI, YOLO) в программные продукты, разворачивает их как микросервисы (FastAPI, Flask, Streamlit, Docker) и оптимизирует инференс (кэширование, similarity cache, батчинг, асинхронная обработка через очереди)	Не знает, что такое similarity cache и ANN	Понимает идею кэширования эмбедингов, но не может интегрировать FAISS; делает синхронные запросы к модели	Реализует similarity cache на основе косинусной близости через FAISS (индекс FlatL2), добавляет простое батчирование (накопление запросов за 100 мс) и асинхронную очередь (RabbitMQ или Redis Stream)	Реализует ANN-индекс (IVF, HNSW) для масштабирования, батчирование с динамическим размером и таймаутом, асинхронный пайплайн инференса (очередь + воркеры + кэш), similarity cache с обновлением по LRU
---	---	--	---	--

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕРКИ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
1.	2	Какая основная проблема классического хеширования $\text{hash}(\text{key}) \% N$ при изменении количества узлов в распределённом кэше? 1) Неравномерное распределение ключей 2) Лавинное перемещение всех ключей при добавлении/удалении узла 3) Невозможность использования строковых ключей 4) Медленный поиск	ПК-2
2.	1,3	Какие элементы используются в алгоритме Consistent Hashing для улучшения равномерности распределения? (Выберите все верные) 1) Кольцевое хеш-пространство (от 0 до $2^{32}-1$) 2) Двоичная куча 3) Виртуальные узлы (multiple virtual nodes per physical node) 4) Алгоритм Raft	ПК-2
3.	3	Чему равна временная сложность поиска узла для ключа в Consistent Hashing при использовании отсортированного списка виртуальных узлов? 1) $O(1)$ 2) $O(\log N)$, где N – число физических узлов 3) $O(\log V)$, где V – число виртуальных узлов (обычно $V = N \times 100 \dots 200$) 4) $O(N)$	ПК-3
4.	4	Какой алгоритм балансировки выбирает для каждого ключа узел с максимальным значением $\text{hash}(\text{key}, \text{node})$, не используя кольцо? 1) Weighted Round Robin 2) Power of Two Choices 3) Consistent Hashing 4) Rendezvous Hashing (Highest Random Weight)	ПК-2
5.	2	Какое преимущество у алгоритма Power of Two Choices перед детерминированной балансировкой? 1) Требуется $O(1)$ памяти 2) Экспоненциально снижает максимальную нагрузку на узел за счёт выбора менее загруженного из двух случайных узлов	ПК-2

		3) Гарантирует равномерное распределение без случайности 4) Работает без согласованности	
6.	3	Какой алгоритм обнаружения отказов использует вероятностную модель времени отклика и вычисляет $\phi = -\log_{10}(P(\text{задержка}))$ для принятия решения? 1) TCP keepalive 2) Gossip-протокол 3) Phi Accrual Detection 4) Raft heartbeat	ПК-3
7.	1,4	Какие из перечисленных протоколов относятся к алгоритмам распределённого консенсуса? (Выберите все верные) 1) Raft 2) HTTP/3 3) AMQP 4) Paxos	ПК-2
8.	2	Какой алгоритм вытеснения кэша удаляет элемент, к которому дольше всего не было обращений, и реализуется с помощью двусвязного списка + хеш-таблицы? 1) LFU 2) LRU 3) FIFO 4) ARC	ПК-3
9.	4	Какая структура данных используется в TinyLFU для приближённого подсчёта частоты обращений с контролируемой ошибкой? 1) Блум-фильтр 2) HyperLogLog 3) Фильтр Блума 4) Count-Min Sketch	ПК-8 (алгоритмические основы ML)
10.	2	Какой алгоритм кэширования считается оптимальным (оффлайн) и вытесняет элемент, который будет использован в самый дальний момент в будущем? 1) LRU 2) Belady's optimal 3) 2Q 4) LFU	ПК-2

11.	3	Какой алгоритм ограничения частоты запросов (rate limiting) хранит только одно значение – время следующего разрешённого запроса (TAT) – и не требует истории? 1) Token Bucket 2) Sliding Window Log 3) GCRA (Generic Cell Rate Algorithm) 4) Leaky Bucket	ПК-3
12.	1	Какой параметр в алгоритме Token Bucket определяет максимальную пиковую нагрузку (burst)? 1) B – ёмкость ведра (bucket size) 2) r – скорость пополнения токенами 3) T – период между запросами 4) τ – толерантность	ПК-3
13.	4	Согласно CAP-теореме, в распределённой системе с разделением (partition) можно гарантировать одновременно только два из трёх свойств. Какие? 1) Consistency, Availability, Latency 2) Consistency, Scalability, Partition tolerance 3) Availability, Durability, Partition tolerance 4) Consistency, Availability, Partition tolerance	ПК-2
14.	2	Какое условие кворума гарантирует, что при чтении данных из распределённого хранилища будет найдена хотя бы одна актуальная версия (при $w + r > n$)? 1) $w > n/2, r > n/2$ 2) $w + r > n$, где n – число реплик 3) $w = r = n$ 4) $w = 1, r = n$	ПК-2
15.	3	Что такое векторные часы (Vector Clocks) в распределённых системах? 1) Глобальное время синхронизации узлов 2) Механизм сжатия логов 3) Структура данных для отслеживания причинно-следственных связей между событиями без глобальных часов 4) Алгоритм выбора лидера	ПК-3
16.	1	Какое семейство CRDT позволяет сходиться к согласованному состоянию без координации за счёт обмена полным состоянием между узлами?	ПК-2

		1) CmRDT (State-based) 2) CvRDT (Operation-based) 3) G-Counter 4) PN-Counter	
17.	2,4	Какие алгоритмы распределённых транзакций используют блокировку ресурсов на этапе подготовки? (Выберите все верные) 1) Saga (хореография) 2) Two-Phase Commit (2PC) 3) Raft 4) Three-Phase Commit (3PC)	ПК-2
18.	3	Какой алгоритм повторных попыток вычисляет задержку как $\min(\max_delay, initial * 2^{\text{attempt}}) + \text{random}()$? 1) Linear backoff 2) Constant backoff 3) Exponential backoff with jitter 4) Fibonacci backoff	ПК-3
19.	4	В каком состоянии Circuit Breaker разрешает небольшое количество запросов для проверки восстановления сервиса? 1) Closed 2) Open 3) Falling 4) Half-Open	ПК-3
20.	2	Какой алгоритм интеграции ML-моделей возвращает кэшированный результат для похожего, но не идентичного входа, на основе близости векторных эмбеддингов? 1) Exact match caching 2) Similarity caching 3) TTL caching 4) Write-through caching	ПК-8
21.	1	Какая метрика расстояния чаще всего используется для сравнения эмбеддингов в similarity caching для текстов и изображений? 1) Косинусная близость (cosine similarity) 2) Евклидово расстояние	ПК-8

		3) Расстояние Манхэттена 4) Коэффициент Жаккара	
22.	3	Что такое dynamic batching в асинхронном инференсе ML-моделей? 1) Запуск модели на каждом запросе отдельно 2) Фиксированный размер батча 3) Накопление запросов до достижения размера батча или таймаута для увеличения пропускной способности GPU 4) Отбрасывание запросов при перегрузке	ПК-8
23.	2	Какой протокол распространения состояния в кластере использует периодический случайный обмен информацией между узлами со сходимостью за $O(\log N)$ раундов? 1) Raft 2) Gossip protocol 3) Paxos 4) Two-Phase Commit	ПК-3
24.	4	В чём основное отличие Weighted Round Robin (WRR) от Smooth Weighted Round Robin (SWRR) в Nginx? 1) SWRR не поддерживает веса 2) WRR поддерживает только целые веса 3) SWRR использует случайность, а WRR – нет 4) SWRR обеспечивает более гладкое распределение без пиковых нагрузок на один узел	ПК-3
25.	1	Какая временная сложность операции get в правильно реализованном LRU-кэше (с хеш-таблицей и двусвязным списком)? 1) $O(1)$ в среднем 2) $O(\log n)$ 3) $O(n)$ 4) $O(n^2)$	ПК-3
26.	2	Какой алгоритм кэширования использует четыре списка (T1, B1, T2, B2) и адаптивный параметр p для баланса между LRU и LFU? 1) 2Q 2) ARC (Adaptive Replacement Cache) 3) TinyLFU	ПК-2

		4) FIFO	
27.	3	Что такое «doorkeeper» в TinyLFU? 1) Механизм шифрования ключей 2) Фильтр для частых ключей 3) Фильтр Блума, защищающий кэш от однократных запросов (one-hit wonders) 4) Очередь входящих ключей	ПК-8
28.	1	Какой алгоритм rate limiting наиболее точен, но требует хранения временных меток всех запросов за окно? 1) Sliding Window Log 2) Fixed Window 3) Token Bucket 4) GCRA	ПК-3
29.	4	Что означает аббревиатура PACELC в распределённых системах? 1) Partition, Availability, Consistency, Eventual, Latency, Consistency 2) Performance, Atomicity, Consistency, Elasticity, Latency, Consistency 3) Partition, Atomicity, Consistency, Elasticity, Latency, C 4) Even when there is a Partition (P), trade-off between C and A; Else (E), trade-off between L (latency) and C	ПК-2
30.	2	Какая гарантия даётся алгоритмом Raft при выборе лидера? 1) Лидером становится узел с наибольшим номером лога 2) Выбранный лидер имеет полностью актуальный журнал (свойство безопасности) 3) Лидер назначается администратором 4) Лидер может быть только один раз	ПК-3
31.	1	Что такое «компенсирующая транзакция» в паттерне Saga? 1) Операция, отменяющая эффект предыдущего шага при ошибке 2) Транзакция с двухфазной фиксацией 3) Транзакция, не требующая отката 4) Транзакция с уровнем изоляции Serializable	ПК-2
32.	3	Какой метод разрешения конфликтов в распределённых системах просто выбирает последнюю запись по времени (часто с использованием физического времени) и используется в Cassandra? 1) First Write Wins	ПК-3

		2) Vector Clocks 3) Last Write Wins (LWW) 4) CRDT	
33.	2	Какой алгоритм обеспечивает отказоустойчивость за счёт репликации журнала и кворума при записи (Raft, Paxos)? 1) Master-slave replication 2) Consensus algorithm 3) Multi-master replication без координации 4) Eventual consistency	ПК-2
34.	4	Какой параметр в алгоритме Exponential Backoff помогает избежать эффекта «льювеного стада» (thundering herd) при массовом пробуждении клиентов? 1) Максимальное число попыток 2) Начальная задержка 3) Экспоненциальный множитель 4) Jitter (случайная добавка)	ПК-3
35.	1	Какое из утверждений о Count-Min Sketch верно? 1) Позволяет приближённо подсчитать частоту элементов с гарантированной верхней границей ошибки при заданном размере 2) Даёт точный подсчёт частоты 3) Не требует памяти 4) Работает только для строк	ПК-8
36.	2	Какая структура данных чаще всего используется для приближённого поиска ближайших соседей (ANN) в векторных базах данных для similarity caching? 1) B-дерево 2) HNSW (Hierarchical Navigable Small World), IVF, или Annoy 3) Хеш-таблица 4) Красно-чёрное дерево	ПК-8
37.	3	Какой показатель качества кэша (hit ratio) для ML-инференса может улучшить similarity caching по сравнению с exact-match кэшем? 1) Только для точных совпадений 2) Ухудшает hit ratio 3) Увеличивает hit ratio за счёт возврата ответа для похожих, но не идентичных запросов	ПК-8

		4) Не влияет	
38.	1	Какая архитектура асинхронного инференса использует очередь сообщений (Kafka, RabbitMQ) и воркеры, обрабатывающие батчи? 1) Decoupled async batch inference 2) Синхронный инференс с балансировкой 3) In-process cache 4) Direct pass-through	ПК-8
39.	2	Какой алгоритм распределения нагрузки обычно используется в Redis Cluster для распределения ключей по слотам (16384 слота)? 1) Rendezvous Hashing 2) Consistent Hashing (модифицированный – хеш-слоты) 3) Weighted Round Robin 4) Random	ПК-2
40.	3	Что такое «виртуальные узлы» (virtual nodes) в Consistent Hashing? 1) Физические узлы с меньшим весом 2) Узлы, работающие в тестовом режиме 3) Множественные хеш-точки одного физического узла, улучшающие равномерность распределения 4) Узлы, не участвующие в распределении	ПК-2
41.	1	Какое математическое ожидание количества ключей, перемещаемых при добавлении одного узла в Consistent Hashing с виртуальными узлами? 1) $\sim K / (N+1)$, где K – общее число ключей, N – старых узлов 2) Все ключи 3) Ключи только одного узла 4) Ни одного	ПК-2
42.	2	Какой алгоритм рекомендуется для rate limiting в распределённой системе, когда нужна строгая точность и низкая память, а также поддержка burst? 1) Fixed window + Redis 2) GCRA с централизованным Redis (через LUA) 3) Leaky bucket 4) Token bucket	ПК-3
43.	3	Какой механизм в Raft предотвращает «расщепление голосования» (split vote) при выборе лидера?	ПК-3

		1) Таймауты выборов 2) Инкремент номера термина 3) Randomised election timeouts 4) Кворум	
44.	1	Какая структура данных используется в реализации 2Q (Two Queues) для защиты от «одноразовых» запросов? 1) FIFO-очередь входящих элементов (Am – допуск) 2) LIFO-стек 3) Приоритетная очередь 4) Двоичная куча	ПК-2
45.	4	Какое значение угла (cosine similarity) обычно считается высокой степенью схожести эмбедингов в similarity caching? 1) 0.5 2) 0.7 3) 0.8 4) 0.95 и выше	ПК-8
46.	2	Какая сложность поиска в отсортированном списке виртуальных узлов в Consistent Hashing? 1) $O(1)$ 2) $O(\log V)$ 3) $O(V)$ 4) $O(N)$	ПК-3
47.	3	Что такое «G-Counter» (Grow-Only Counter) в семействе CRDT? 1) Счётчик, который может увеличиваться и уменьшаться 2) Счётчик только для уменьшения 3) Счётчик, который может только увеличиваться, конфликтов не возникает 4) Счётчик с координацией	ПК-2
48.	1	Какой паттерн отказоустойчивости сочетает в себе Exponential Backoff и Circuit Breaker? 1) Retry with circuit breaker 2) Bulkhead 3) Timeout only 4) Cache aside	ПК-3

49.	4	В чём преимущество адаптивного таймаута (adaptive timeout) по сравнению с фиксированным? 1) Проще реализовать 2) Всегда выбирает меньший таймаут 3) Игнорирует задержки сети 4) Динамически подстраивается под реальные проценты задержки (p95, p99)	ПК-3
50.	2	Какая проблема «горячих ключей» (hot keys) в распределённом кэше решается репликацией ключа на несколько узлов? 1) Неравномерное хеширование 2) Сверхвысокая нагрузка на один узел из-за множества запросов к одному ключу 3) Потеря данных при отказе 4) Высокая задержка при записи	ПК-2
51.	2	Какой алгоритм распределения нагрузки гарантирует, что распределение запросов будет строго пропорционально весам, но при этом обеспечивает «гладкость» (отсутствие длинных серий одинаковых выборов)? 1) Weighted Round Robin 2) Smooth Weighted Round Robin (SWRR) из Nginx 3) Least Connections 4) Power of Two Choices	ПК-3
52.	1	Какая временная сложность у алгоритма Rendezvous Hashing без оптимизаций? 1) $O(N)$, где N – число узлов 2) $O(\log N)$ 3) $O(1)$ 4) $O(N^2)$	ПК-2
53.	3	Какой протокол использует «heartbeats» и «AppendEntries» RPC для репликации журнала? 1) Gossip 2) Paxos 3) Raft 4) Two-Phase Commit	ПК-3
54.	1,4	Какие из перечисленных алгоритмов являются онлайн-алгоритмами кэширования (не имеют информации о будущих запросах)? (Выберите все верные)	ПК-2

		1) LRU 2) Belady's optimal 3) Offline algorithm 4) LFU	
55.	2	Какой параметр в ARC определяет, сколько элементов отдаётся под «недавние» (LRU) и сколько под «частые» (LFU)? 1) Размер кэша 2) Адаптивный параметр p (target size for T1) 3) Коэффициент ageing 4) Размер списка B1	ПК-2
56.	3	Что такое «SLAB-аллокатор» в Memcached? 1) Алгоритм распределения запросов 2) Механизм репликации 3) Способ управления памятью с разбиением на классы размеров для предотвращения фрагментации 4) Протокол кластеризации	ПК-3
57.	1	Какой алгоритм rate limiting использует очередь фиксированной длины и обрабатывает запросы с постоянной скоростью, не допуская burst? 1) Leaky Bucket 2) Token Bucket 3) GCRA 4) Sliding Window	ПК-3
58.	4	Как в GCRA вычисляется новое значение TAT (Time of Next Allowed Request) при разрешении запроса? 1) $TAT = now$ 2) $TAT = TAT + T$ 3) $TAT = \max(now, TAT) + T$ 4) $TAT = \max(now, TAT) + T$, где T – период между запросами	ПК-3
59.	2	Какое свойство системы нарушается при выборе Availability и Partition Tolerance в CAP-теореме? 1) Partition Tolerance 2) Consistency (линейная согласованность) 3) Availability	ПК-2

		4) Durability	
60.	3	Что такое «eventual consistency»? 1) Сильная согласованность после каждой записи 2) Отсутствие согласованности 3) Гарантия, что при отсутствии новых обновлений все реплики сойдутся к одному значению за конечное время 4) Согласованность только для чтения	ПК-2
61.	1	Какая структура данных используется в CRDT для счётчика, который можно и увеличивать, и уменьшать (PN-Counter)? 1) Два G-Counter'a: один для инкрементов, другой для декрементов 2) Один счётчик со знаком 3) Вектор часов 4) Реплицированный лог	ПК-3
62.	2	В чём основное отличие Saga с оркестрацией от Saga с хореографией? 1) Хореография требует центрального координатора, оркестрация – нет 2) Оркестрация имеет центральный координатор, хореография – децентрализованное взаимодействие через события 3) Хореография использует двухфазную фиксацию 4) Оркестрация не поддерживает компенсации	ПК-2
63.	4	Какой алгоритм таймаута рекомендуется использовать в распределённых системах для адаптации к изменяющимся сетевым задержкам? 1) Фиксированный таймаут (5000 ms) 2) Таймаут по умолчанию из библиотеки 3) Бесконечный таймаут 4) Адаптивный таймаут на основе процентилях (p95, p99) измеренных задержек	ПК-3
64.	1	Что такое «fallback» в контексте отказоустойчивости? 1) Возврат запасного результата (кэшированные данные, значение по умолчанию) при ошибке основного вызова 2) Повторная попытка вызова 3) Разрыв цепи 4) Логирование ошибки	ПК-3
65.	2,3	Какие из перечисленных библиотек реализуют алгоритмы отказоустойчивости (Circuit Breaker, Retry, Bulkhead) для Java? (Выберите все верные)	ПК-3

		1) Hibernate 2) Resilience4j 3) Netflix Hystrix (устаревший) 4) Log4j	
66.	4	Какой метод приближённого поиска ближайших соседей (ANN) используется в библиотеке FAISS от Facebook? 1) KD-tree 2) Блум-фильтр 3) LSH (Locality Sensitive Hashing) 4) IVF (Inverted File Index) с квантованием	ПК-8
67.	1	Для чего используется косинусная близость в similarity caching? 1) Для сравнения векторных эмбеддингов: чем ближе к 1, тем более похожи объекты 2) Для шифрования эмбеддингов 3) Для сжатия векторных представлений 4) Для оценки частоты запросов	ПК-8
68.	3	Какой подход к интеграции ML-моделей позволяет обрабатывать запросы группами на GPU, увеличивая пропускную способность ценой увеличения задержки? 1) Синхронный инференс (один запрос на модель) 2) Предвычисление эмбеддингов оффлайн 3) Динамическое батчирование (dynamic batching) 4) Кэширование точных совпадений	ПК-8
69.	2	Какая проблема решается в алгоритме 2Q (Two Queues) с помощью очереди Am (недавно поступившие элементы)? 1) Высокое потребление памяти 2) Защита от загрязнения кэша одноразовыми запросами (one-hit wonders) 3) Ускорение поиска 4) Репликация	ПК-2
70.	4	Что такое «рабочий набор» (working set) в теории кэширования? 1) Набор элементов, которые никогда не вытесняются 2) Набор элементов, загруженных при старте 3) Набор всех элементов в кэше	ПК-2

		4) Набор страниц/данных, активно используемых в текущий период времени; чем выше попадание рабочего набора в кэш, тем выше hit ratio	
71.	1	Какой алгоритм вытеснения кэша использует «приближённое» LFU с помощью Count-Min Sketch и doorkeeper? 1) TinyLFU 2) ARC 3) 2Q 4) LIRS	ПК-8
72.	3	Какую команду Redis нужно выполнить для настройки политики вытеснения «allkeys-lru»? 1) CONFIG SET maxmemory-policy volatile-lru 2) CONFIG SET eviction-policy lru 3) CONFIG SET maxmemory-policy allkeys-lru 4) CONFIG SET policy lru-all	ПК-3
73.	2	Какой из алгоритмов распределения нагрузки требует вычисления хеша для каждого узла на каждый запрос, что приводит к сложности $O(N)$ на запрос (без кэширования)? 1) Consistent Hashing 2) Rendezvous Hashing (без кэширования результатов) 3) Power of Two Choices 4) Weighted Round Robin	ПК-2
74.	1	В чём преимущество Rendezvous Hashing перед Consistent Hashing при малом числе узлов? 1) Потенциально более равномерное распределение без виртуальных узлов 2) Лучшая производительность поиска 3) Меньшее потребление памяти 4) Более простая реализация	ПК-2
75.	4	Какой параметр в Phi Accrual Detection определяет порог, после которого узел считается отказавшим? 1) Время ожидания heartbeat 2) Количество пропущенных heartbeat 3) Стандартное отклонение 4) Значение ϕ (phi), обычно от 8 до 12	ПК-3

76.	2	Что такое «consensus» в распределённых системах? 1) Синхронизация времени всех узлов 2) Договорённость множества узлов об одном значении (или последовательности команд) в условиях отказов 3) Общая память для всех узлов 4) Блокировка ресурсов	ПК-2
77.	3	Какой алгоритм распределённого консенсуса считается более понятным для реализации, чем Paxos, и разбит на три подзадачи: выбор лидера, репликация лога, безопасность? 1) Zab 2) Viewstamped Replication 3) Raft 4) Multi-Paxos	ПК-2
78.	1	Какое свойство Raft гарантирует, что лидер имеет все зафиксированные записи из предыдущих терминов? 1) Свойство безопасности (Election Safety) 2) Availability 3) Linearizability 4) Eventual consistency	ПК-2
79.	4	Какой алгоритм ограничения частоты запросов чаще всего используется в облачных API (AWS, Google) благодаря балансу между точностью и использованием памяти? 1) Fixed Window 2) Leaky Bucket 3) Token Bucket только 4) Sliding Window (вариант счётчика) или GCRA в зависимости от провайдера	ПК-3
80.	2	Что такое «адаптивный таймаут»? 1) Таймаут, задаваемый вручную администратором 2) Таймаут, автоматически вычисляемый на основе недавней истории времени отклика (например, $\text{median} + 2 * \text{MAD}$) 3) Таймаут, равный максимальной задержке в системе 4) Таймаут, отключаемый при высокой нагрузке	ПК-3

81.	3	Какой тип CRDT обеспечивает сходимость только при передаче операций (commutative, idempotent) и требует надёжной доставки? 1) CmRDT (state-based) 2) G-Counter 3) CvRDT (operation-based) 4) PN-Counter	ПК-2
82.	1	Что такое «векторный эмбединг» в контексте ML? 1) Числовое представление объекта (изображения, текста, пользователя) в многомерном пространстве признаков 2) Бинарный код объекта 3) Хеш объекта 4) Строковый идентификатор	ПК-8
83.	2,4	Какие библиотеки используются для построения индексов ANN (приближённого поиска ближайших соседей)? (Выберите все верные) 1) TensorFlow 2) FAISS 3) PyTorch 4) Annoy	ПК-8
84.	3	Какой подход в similarity caching называется «кэширование по центроиду» (centroid caching)? 1) Кэширование каждого уникального эмбединга отдельно 2) Кэширование только точных совпадений 3) Группировка похожих эмбедингов вокруг центроида и кэширование одного результата для группы 4) Кэширование только самых частых запросов	ПК-8
85.	1	Какой алгоритм распределения нагрузки используется в Amazon DynamoDB? 1) Consistent Hashing с виртуальными узлами 2) Rendezvous Hashing 3) Random 4) Round Robin	ПК-2
86.	2	Какая теоретическая нижняя граница конкурентного соотношения для онлайн-алгоритмов кэширования? 1) 1	ПК-2

		2) k (размер кэша) для детерминированных алгоритмов 3) $\log k$ 4) k^2	
87.	4	Что означает параметр maxmemory-policy volatile-ttl в Redis? 1) Вытеснение любых ключей по LRU 2) Вытеснение только ключей с истекающим TTL по LRU 3) Вытеснение любых ключей по LFU 4) Вытеснение ключей с TTL, у которых осталось меньше всего времени жизни	ПК-3
88.	3	Какой алгоритм rate limiting является наиболее точным, но требует O(окно) памяти и часто используется в точных системах биллинга? 1) Token Bucket 2) GCRA 3) Sliding Window Log (хранит все метки времени) 4) Fixed Window	ПК-3
89.	2	В чём отличие Leaky Bucket от Token Bucket? 1) Leaky Bucket позволяет burst, а Token Bucket – нет 2) Leaky Bucket не допускает burst (запросы обрабатываются с постоянной скоростью), а Token Bucket допускает 3) Token Bucket требует больше памяти 4) Leaky Bucket точнее	ПК-3
90.	1	Что такое «Sketch» в алгоритмах приближённого подсчёта (Count-Min Sketch)? 1) Вероятностная структура данных, использующая несколько хеш-функций и матрицу счётчиков 2) Точный счётчик 3) Блум-фильтр 4) HyperLogLog	ПК-8
91.	3	Какое конкурентное соотношение (competitive ratio) имеет LRU-кэш по сравнению с оптимальным алгоритмом Беладии? 1) 1 2) $\log k$ 3) k (при размере кэша k) в худшем случае, но на практике лучше 4) k^2	ПК-2

92.	2	Какой алгоритм репликации в распределённых БД использует «синхронную» репликацию с гарантией, что запись подтверждается только после записи на w узлах? 1) Асинхронная репликация 2) Quorum-based репликация ($w + r > n$) 3) Chain replication 4) Single-master	ПК-2
93.	4	Что такое «RPO» (Recovery Point Objective) в контексте распределённых систем? 1) Время восстановления системы 2) Целевое время простоя 3) Максимальная задержка репликации 4) Максимально допустимый объём потерянных данных при сбое (время, на которое можно откатиться)	ПК-2
94.	1	Какой алгоритм используется в библиотеке Caffeine для высокопроизводительного кэширования в Java? 1) Window TinyLFU 2) ARC 3) 2Q 4) LIRS	ПК-3
95.	2	Какой метод разрешения конфликтов в распределённых системах основан на частичном порядке событий и требует внешнего разрешения при несравнимых версиях? 1) Last Write Wins 2) Vector Clocks 3) First Write Wins 4) CRDT с join	ПК-3
96.	3	В чём основное отличие Phi Accrual Detection от простого таймаута? 1) Использует фиксированное время ожидания 2) Не требует истории откликов 3) Адаптируется к изменяющейся задержке сети через вероятностную модель и динамический порог ϕ 4) Требуется синхронизации часов	ПК-3

97.	2	Какой параметр в Exponential Backoff уменьшает вероятность синхронных повторных попыток от множества клиентов? 1) Initial delay 2) Jitter (случайная добавка) 3) Multiplier 4) Max attempts	ПК-3
98.	1	Что такое «хореография» (choreography) в паттерне Saga? 1) Каждый сервис публикует события, другие сервисы реагируют и публикуют свои события; нет центрального координатора 2) Есть центральный оркестратор, который вызывает сервисы по порядку 3) Все сервисы блокируются до завершения транзакции 4) Используется двухфазная фиксация	ПК-2
99.	3	Какой алгоритм следует использовать для similarity caching, если требуется масштабирование до миллионов векторов и под-секундный поиск? 1) Полный перебор (brute-force) по всем векторам 2) В-дерево по эмбедингам 3) Индекс ANN (например, HNSW, IVF) + exact search для небольшой кандидатной выборки 4) Хеш-таблица по округлённым эмбедингам	ПК-8
100.	4	Какая из перечисленных библиотек НЕ является библиотекой приближённого поиска ближайших соседей? 1) FAISS 2) Annoy 3) ScaNN 4) TensorFlow (это библиотека для ML, не специализированная для ANN)	ПК-8

2. Описание шкалы оценивания

Оценка успеваемости студентов по дисциплине осуществляется в ходе текущего контроля и промежуточной аттестации. Для проверки уровня сформированности компетенций используется совокупность оценочных средств, позволяющих оценить знания, умения и навыки, приобретённые студентами в процессе изучения дисциплины.

Основными формами текущего контроля являются:

- выполнение и защита лабораторных работ;
- устные опросы (собеседования) по темам занятий;
- проверка индивидуальных заданий;
- тестирование по отдельным разделам дисциплины.

Промежуточная аттестация проводится в форме экзамена и включает выполнение практического задания и/или ответы на теоретические вопросы.

3. Критерии оценивания компетенций заданий открытого типа

Оценка «отлично» выставляется студенту, если являются верными результаты демонстрации знаний об основных структурах данных (массивы, связанные списки, стеки, очереди, хеш-таблицы, деревья (AVL, красно-чёрные, В-деревья), графы, кучи), алгоритмах (сортировки, поиска, обхода графов (BFS, DFS), алгоритмы на графах (Дейкстра, Флойда–Уоршелла, минимальное остовное дерево), рекурсия, динамическое программирование, алгоритмы с возвратом, алгоритмы сжатия, поиск подстрок), методах оценки сложности (О-нотация, амортизационный анализ, анализ по памяти); а также знаний и умений в области проектирования архитектуры программных систем среднего и крупного масштаба с выбором оптимальных структур данных и алгоритмов, применении современных паттернов проектирования (итератор, стратегия, посетитель, фабрика), владении навыками реализации алгоритмов на современных языках программирования (C/C++, Python, Java и др.), создании консольных и desktop-приложений, работе с базами данных (индексирование, оптимизация запросов с использованием структур данных); способности адаптировать и применять методы искусственного интеллекта и машинного обучения (например, алгоритмы кластеризации (k-means), регрессии, деревья решений, интеграция готовых моделей из scikit-learn или TensorFlow Lite для задач классификации и предсказания, реализация базовых алгоритмов ИИ, таких как поиск в пространстве состояний, генетические алгоритмы) для решения профессиональных задач; представленный отчёт (программный код, пояснения, блок-схемы, результаты замеров производительности, примеры интеграции моделей ИИ) не содержит ошибок, решены все задачи лабораторных работ и индивидуальных заданий. Компетенции **ПК-2, ПК-3, ПК-8** освоены на **высоком уровне**.

Оценка «хорошо» выставляется студенту, если являются верными результаты демонстрации указанных знаний, но отчётный документ содержит незначительные ошибки, не влияющие на общий результат (например, неточности в оценке сложности, отсутствие некоторых комментариев, неоптимальная интеграция ИИ-модели), либо не представлены отдельные второстепенные задачи, не влияющие на общий результат. Компетенции **ПК-2, ПК-3, ПК-8** освоены на **среднем уровне**.

Оценка «удовлетворительно» выставляется студенту, если в основном верными являются результаты демонстрации знаний, но отчётный документ содержит ошибки, не позволяющие однозначно оценить качество разработки (например, неверная реализация ключевой структуры данных, неправильный выбор алгоритма, ошибки в адаптации метода ИИ); представлены не все данные, задачи решены не в полном объёме. Компетенции **ПК-2, ПК-3, ПК-8** освоены на **минимальном уровне**.

Оценка «неудовлетворительно» выставляется студенту, если являются неверными результаты демонстрации знаний, отчётный документ содержит много грубых ошибок (непонимание базовых структур и алгоритмов, код не компилируется или не решает задачу, отсутствие понимания методов ИИ), не представлены ключевые данные, задачи не решены. Компетенции **ПК-2, ПК-3, ПК-8** не сформированы.

4. Критерии оценивания компетенций заданий закрытого типа

Оценка «отлично» выставляется студенту, если он правильно ответил на 88-100% от общего количества заданий. Компетенция ПК-2 освоена на высоком уровне.

Оценка «хорошо» выставляется студенту, если он правильно ответил на 72-87% от общего количества заданий. Компетенция ПК-2 освоена на среднем уровне.

Оценка «удовлетворительно» выставляется студенту, если он правильно ответил на 53-71% от общего количества заданий. Компетенция ПК-2 освоена на минимальном уровне.

Оценка «неудовлетворительно» выставляется студенту, выставляется студенту, если он правильно ответил на менее 53% от общего количества заданий. Компетенция ПК-2 не сформирована.

Оценка «отлично» выставляется студенту, если он правильно ответил на 88-100% от общего количества заданий. Компетенция ПК-3 освоена на высоком уровне.

Оценка «хорошо» выставляется студенту, если он правильно ответил на 72-87% от общего количества заданий. Компетенция ПК-3 освоена на среднем уровне.

Оценка «удовлетворительно» выставляется студенту, если он правильно ответил на 53-71% от общего количества заданий. Компетенция ПК-3 освоена на минимальном уровне.

Оценка «неудовлетворительно» выставляется студенту, выставляется студенту, если он правильно ответил на менее 53% от общего количества заданий. Компетенция ПК-3 не сформирована.

Оценка «отлично» выставляется студенту, если он правильно ответил на 88-100% от общего количества заданий. Компетенция ПК-8 освоена на высоком уровне.

Оценка «хорошо» выставляется студенту, если он правильно ответил на 72-87% от общего количества заданий. Компетенция ПК-8 освоена на среднем уровне.

Оценка «удовлетворительно» выставляется студенту, если он правильно ответил на 53-71% от общего количества заданий. Компетенция ПК-8 освоена на минимальном уровне.

Оценка «неудовлетворительно» выставляется студенту, выставляется студенту, если он правильно ответил на менее 53% от общего количества заданий. Компетенция ПК-8 не сформирована.