

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего
образования
«ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»

Структурное подразделение «Сибирская школа геонаук (119)»

УТВЕРЖДЕНА:
на заседании ДОТ
Протокол №40 от 13 мая 2026 г.

Рабочая программа дисциплины

«АРХИТЕКТУРА ЭВМ И СИСТЕМ / COMPUTER AND SYSTEM ARCHITECTURE»

Направление: 09.03.02 Информационные системы и технологии

Информационные технологии в науках о Земле и окружающей среде / Information
Technologies in Earth and Environmental Sciences

Квалификация: Бакалавр

Форма обучения: очная

Документ подписан простой
электронной подписью
Составитель программы:
Гантимурова Светлана
Анатольевна
Дата подписания: 18.06.2026

Документ подписан простой
электронной подписью
Утвердил: Ланько Анна
Викторовна
Дата подписания: 18.06.2026

Документ подписан простой
электронной подписью
Согласовал: Паршин
Александр Вадимович
Дата подписания: 18.06.2026

Год набора – 2026

Иркутск, 2026 г.

1 Перечень планируемых результатов обучения по дисциплине, соотнесённых с планируемыми результатами освоения образовательной программы

1.1 Дисциплина «Архитектура ЭВМ и систем / Computer and System Architecture» обеспечивает формирование следующих компетенций с учётом индикаторов их достижения

Код, наименование компетенции	Код индикатора компетенции
ОПК ОС-7 Способен осуществлять выбор платформ и инструментальных программно-аппаратных средств для реализации информационных систем	ОПК ОС-7.1

1.2 В результате освоения дисциплины у обучающихся должны быть сформированы

Код индикатора	Содержание индикатора	Результат обучения
ОПК ОС-7.1	Способен выполнять оценку архитектуры аппаратных комплексов информационной системы	Знать теоретические основы аппаратной архитектуры, необходимые для анализа комплексов ИС в недропользовании Уметь анализировать производительность процессора, памяти и ИО-систем, а также сравнивать архитектуры ЭВМ для задач обработки геоданных в ИС недропользования. Владеть методиками расчета производительности (MIPS, FLOPS), моделированием совместимости компонентов и интерпретацией спецификаций для оценки архитектуры аппаратных комплексов ИС.

2 Место дисциплины в структуре ООП

Изучение дисциплины «Архитектура ЭВМ и систем / Computer and System Architecture» базируется на результатах освоения следующих дисциплин/практик: Нет

Дисциплина является предшествующей для дисциплин/практик: «Архитектура информационных систем / Information System Architecture»

3 Объем дисциплины

Объем дисциплины составляет – 3 ЗЕТ

Вид учебной работы	Трудоемкость в академических часах (Один академический час соответствует 45 минутам астрономического часа)	
	Всего	Семестр № 2
Общая трудоемкость дисциплины	108	108
Аудиторные занятия, в том числе:	48	48
лекции	16	16

лабораторные работы	32	32
практические/семинарские занятия	0	0
Самостоятельная работа (в т.ч. курсовое проектирование)	60	60
Трудоемкость промежуточной аттестации	0	0
Вид промежуточной аттестации (итогового контроля по дисциплине)	Зачет	Зачет

4 Структура и содержание дисциплины

4.1 Сводные данные по содержанию дисциплины

Семестр № 2

№ п/п	Наименование раздела и темы дисциплины	Виды контактной работы						СРС		Форма текущего контроля
		Лекции		ЛР		ПЗ(СЕМ)				
		№	Кол. Час.	№	Кол. Час.	№	Кол. Час.	№	Кол. Час.	
1	2	3	4	5	6	7	8	9	10	11
1	1.Основы вычислительных систем	1	2	1	4			3	10	Устный опрос
2	2. Архитектура центрального процессора	2	2	2	4			1	10	Устный опрос
3	3. Иерархия памяти	3	2	3, 4	8					Устный опрос
4	4. Системы ввода- вывода и периферия	4	2							Устный опрос
5	5. Базовые шины и интерфейсы	5	2	5	4			5	10	Устный опрос
6	6. Параллелизм и многопроцессорн ые системы			6	6			4	10	Устный опрос
7	7. Производительно сть и оценка систем	6	3	7	6			2	10	Устный опрос
8	8. Конфигурировани е аппаратных комплексов	7	3					2	10	Устный опрос
	Промежуточная аттестация									Зачет
	Всего		16		32				60	

4.2 Краткое содержание разделов и тем занятий

Семестр № 2

№	Тема	Краткое содержание
1	1.Основы вычислительных систем	Изучение фундаментальных понятий электронно-вычислительных машин (ЭВМ), их эволюции от первых компьютеров к современным системам;

		классификация по Флинну (SISD, SIMD, MISD, MIMD) с примерами параллелизма в обработке геофизических данных (сейсмические расчеты); принципы фон Неймана и Гарвардской архитектуры; обзор многоядерных и GPU-систем для задач недропользования, включая оценку масштабируемости для больших объемов геоданных.
2	2. Архитектура центрального процессора	Детальный разбор внутренней структуры ЦП: регистры общего назначения, аккумуляторы, флаговые регистры; арифметико-логическое устройство (АЛУ) и операции над данными (арифметика, логика, сдвиги); устройство управления (микрокод, конечный автомат); конвейеризация (стадии fetch-decode-execute-writeback) и суперскалярность; ветвления и предсказатели; примеры применения в процессорах Intel/AMD/ARM для моделирования геологических процессов.
3	3. Иерархия памяти	Организация памяти как пирамиды: регистры, L1/L2/L3 кэш (ассоциативность, замещение LRU/FIFO); оперативная память (DRAM, SRAM, ECC); виртуальная память (страничная организация, TLB); вторичная память (HDD, SSD, RAID-массивы); принципы локальности (времени и пространства); расчет времени доступа и пропускной способности для хранения больших массивов геофизических данных (сетки, томограммы).
4	4. Системы ввода-вывода и периферия	Классификация устройств ИО (блочные/символьные, DMA-прямой доступ); контроллеры и буферизация; прерывания (векторные, приоритетные); периферия для геоданных (сенсоры сейсмоприборов, GPS, сканеры скважин); примеры интеграции в ИС недропользования; оценка bottlenecks в потоках данных от датчиков к ЦП.
5	5. Базовые шины и интерфейсы	Архитектура шин (адресная, данных, управления); синхронные/асинхронные шины; стандарты PCI/PCIe, USB (версии 2.0/3.0/Type-C), SATA/NVMe; сетевые интерфейсы Ethernet (10/100/1000 Гбит/с), InfiniBand для кластеров; протоколы обмена (DMA over PCIe); применение в подключении геофизического оборудования и сетей передачи данных в реальном времени.
6	6. Параллелизм и многопроцессорные системы	Уровни параллелизма (битовый, инструкционный, потоковый, задачный); симметричные/асимметричные многопроцессорные системы (SMP, NUMA); кластеры и суперкомпьютеры (Beowulf); GPU-вычисления (CUDA/OpenCL) для параллельной

		обработки геоданных; законы Амдала и Густафсона для оценки ускорения в задачах моделирования пластов.
7	7. Производительность и оценка систем	Метрики производительности (MIPS, MFLOPS, CPI, IPC); бенчмарки (SPEC, Linpack); факторы влияния (закон Мура, bottlenecks памяти/ИО); методы профилирования (perf, VTune); расчет эффективности для аппаратных комплексов ИС в недропользовании.
8	8. Конфигурирование аппаратных комплексов	Принципы сборки комплексов ИС: выбор компонентов по задачам (серверы для ГИС, workstations для 3D-моделирования); виртуализация (VMware, Hyper-V); облачные аппаратные платформы (AWS EC2, Azure); оценка совместимости и масштабируемости; кейсы конфигурации для геоинформационных систем недропользования с учетом бюджета и надежности.

4.3 Перечень лабораторных работ

Семестр № 2

№	Наименование лабораторной работы	Кол-во академических часов
1	Лабораторная работа №1: Моделирование классификации ЭВМ по Флинну в Python с использованием NumPy для параллельной обработки простого массива геоданных (симуляция SIMD на векторных операциях).	4
2	Лабораторная работа №2: Симуляция конвейера процессора в Python (стадии fetch-decode-execute) с расчетом CPI для задач арифметической обработки координат скважин.	4
3	Лабораторная работа №3: Измерение локальности доступа в кэше с помощью Python (timeit) на массивах геосеток, расчет hit/miss ratio и времени доступа.	4
4	Лабораторная работа №4: Симуляция DMA и прерываний в Python с многопоточностью (threading) для моделирования потокового ввода геофизических данных от виртуальных сенсоров.	4
5	Лабораторная работа №5: Тестирование сетевых интерфейсов с помощью Python (socket) для передачи фрагмента геоданных, измерение пропускной способности Ethernet-подобных соединений.	4
6	Лабораторная работа №6: Параллельная обработка геоданных с multiprocessing в Python, расчет ускорения по закону Амдала на	6

	симуляции сейсмического моделирования.	
7	Лабораторная работа №7: Профилирование производительности Python-скрипта (cProfile) для задач геообработки, расчет MIPS/FLOPS и выявление bottlenecks.	6

4.4 Перечень практических занятий

Практических занятий не предусмотрено

4.5 Самостоятельная работа

Семестр № 2

№	Вид СРС	Кол-во академических часов
1	Оформление отчетов по лабораторным и практическим работам	10
2	Подготовка к зачёту	20
3	Подготовка к практическим занятиям (лабораторным работам)	10
4	Подготовка к сдаче и защите отчетов	10
5	Проработка разделов теоретического материала	10

В ходе проведения занятий по дисциплине используются следующие интерактивные методы обучения: работа в малых группах

5 Перечень учебно-методического обеспечения дисциплины

5.1 Методические указания для обучающихся по освоению дисциплины

5.1.1 Методические указания для обучающихся по лабораторным работам:

Методические указания к лабораторным работам

Лабораторная работа №1. Моделирование классификации ЭВМ по Флинну

Цель работы: Освоить классификацию вычислительных систем по Флинну и реализовать моделирование SIMD-операций на примере параллельной обработки геоданных в Python с NumPy для оценки базовых принципов архитектуры ЭВМ.

Теоретический справочник:

ЭВМ — электронно-вычислительная машина, устройство для автоматической обработки данных по программе. Классификация Флинна: SISD (Single Instruction Single Data) — одна инструкция на один поток данных (последовательные процессоры); SIMD (Single Instruction Multiple Data) — одна инструкция на несколько потоков данных (векторные процессоры, GPU); MISD (Multiple Instruction Single Data) — несколько инструкций на один поток данных (системы с резервированием для отказоустойчивости); MIMD (Multiple Instruction Multiple Data) — несколько инструкций на несколько потоков данных (многопроцессорные системы). NumPy — библиотека Python для многомерных массивов и векторных вычислений. CPI — Cycles Per Instruction, количество тактовых циклов на одну инструкцию. Speedup — коэффициент ускорения, отношение времени последовательного выполнения к параллельному.

Ход работы:

1. Установить NumPy (pip install numpy).
2. Создать массив геоданных (например, 1D-массив амплитуд 10000 точек).

3. Реализовать последовательную обработку (SISD: цикл for с суммой/нормализацией).
4. Переписать на SIMD (NumPy vectorize: сумма/нормализация массива).
5. Измерить время выполнения (time.time()), рассчитать speedup.
6. Визуализировать результаты (matplotlib plot).

Ожидаемый результат: Отчет с кодом, таблицей времени/SIMD-speedup ($>10x$), графиками обработки геоданных и выводами об эффективности классификации Флинна.

Контрольные вопросы:

1. В чем разница между SISD и SIMD в контексте геоданных?
2. Как NumPy реализует SIMD-подобный параллелизм?
3. Приведите пример MIMD для задач недропользования.

Лабораторная работа №2. Симуляция конвейера процессора

Цель работы: Смоделировать стадии конвейера ЦП (fetch-decode-execute) в Python для расчета CPI при обработке координат скважин и анализа влияния на производительность архитектуры.

Теоретический справочник:

ЦП — центральный процессор, основной вычислительный блок ЭВМ. Конвейер — метод параллельного выполнения инструкций путем разделения на стадии: Fetch (извлечение инструкции из памяти), Decode (декодирование инструкции), Execute (выполнение операции), Writeback (запись результата в регистры). CPI — Cycles Per Instruction, среднее число тактовых циклов на инструкцию (идеальное значение 1). Hazard — конфликт в конвейере: data hazard (зависимость данных), structural hazard (конфликт ресурсов), control hazard (ветвление). АЛУ — арифметико-логическое устройство, блок для выполнения арифметических (+, -), логических (AND, OR) и сдвиговых операций.

Ход работы:

1. Создать класс Pipeline с очередями стадий (deque из collections).
2. Сгенерировать инструкции (список операций над координатами: add, mul, shift).
3. Симулировать 100 циклов: пропускать инструкции по стадиям, обрабатывать hazards (stall).
4. Рассчитать $CPI = \text{циклы} / \text{инструкций}$.
5. Тестировать на данных скважин (массив точек).
6. Вывести логи и графики заполнения конвейера.

Ожидаемый результат: Код симулятора, таблица CPI (1.5-3.0), графики стадий, анализ hazards для геоданных.

Контрольные вопросы:

1. Что вызывает control hazard в конвейере?
2. Какова роль АЛУ в Execute-стадии?
3. Почему $CPI > 1$ в реальных процессорах?

Лабораторная работа №3. Измерение локальности доступа в кэше

Цель работы: Исследовать принципы локальности памяти на массивах геосеток в Python, рассчитать hit/miss ratio и время доступа для оценки иерархии памяти.

Теоретический справочник:

Иерархия памяти — многоуровневая структура: L1-кэш (самый быстрый, 32-64 КБ, на ядре), L2-кэш (256-512 КБ, на ядре), L3-кэш (несколько МБ, общий на процессор).

Локальность времени — повторный доступ к тем же данным в ближайшее время;

локальность пространства — доступ к близко расположенным данным. LRU (Least

Recently Used) — политика замещения, удаляет наименее недавно использованный блок.

Hit ratio — доля успешных обращений в кэш (попадания); miss ratio — доля промахов.

TLB — Translation Lookaside Buffer, быстрый кэш таблиц страниц виртуальной памяти.

Ход работы:

1. Создать 2D-геосетку (numpy array 1000x1000).

2. Реализовать доступы: sequential (row-major), random, strided.
3. Измерить время (timeit.repeat) для каждого паттерна.
4. Симулировать кэш (dict с LRU-eviction, размер 64 блока).
5. Рассчитать hit ratio и среднее время доступа.
6. Построить графики зависимости от размера данных.

Ожидаемый результат: Отчет с замерами (sequential hit>90%, random<20%), графиками, выводами о локальности для геоданных.

Контрольные вопросы:

1. В чем разница локальности времени и пространства?
2. Как LRU работает в кэше?
3. Почему row-major быстрее column-major?

Лабораторная работа №4. Симуляция DMA и прерываний

Цель работы: Смоделировать DMA и прерывания с многопоточностью в Python для потокового ввода геофизических данных и выявления bottlenecks ИО.

Теоретический справочник:

ИО — ввод-вывод, обмен данными между ЦП и периферией. Блочные устройства — передают данные блоками (диски); символьные — по символам (клавиатура). DMA (Direct Memory Access) — прямой доступ периферии к памяти без участия ЦП.

Прерывание — асинхронный сигнал от устройства к ЦП для обработки события (IRQ — Interrupt Request, вектор — номер обработчика). Threading — модуль Python для создания потоков выполнения. Буферизация — временное хранение данных в очередях для сглаживания скоростей.

Ход работы:

1. Создать потоки: Sensor (генератор данных), DMA (копирует в буфер), CPU (обработчик).
2. Использовать threading.Thread и Queue для буфера.
3. Симулировать прерывания (Event для уведомлений).
4. Измерить throughput (данные/сек), latency.
5. Тестировать на 1М точек сенсоров.
6. Проанализировать bottlenecks (time.sleep моделирует).

Ожидаемый результат: Код, графики throughput/latency, таблица сравнения CPU-poll vs DMA+IRQ.

Контрольные вопросы:

1. В чем преимущество DMA над PIO?
2. Как работают приоритетные прерывания?
3. Почему многопоточность в Python имеет GIL?

Лабораторная работа №5. Тестирование сетевых интерфейсов

Цель работы: Протестировать пропускную способность сетевых интерфейсов Python socket для передачи геоданных и оценки интерфейсов в архитектуре ЭВМ.

Теоретический справочник:

Шина — набор проводников для передачи сигналов: адресная (указывает адрес), данных (передает байты), управления (сигналы готовности/запроса). Синхронная шина — тактируется общим clock; асинхронная — по рукопожатию. Ethernet — локальная сеть по стандарту IEEE 802.3 (скорости 10/100/1000 Мбит/с, CSMA/CD — контроль доступа).

Socket — программный интерфейс для сетевого обмена (TCP — надежный, UDP — без гарантии). PCIe — Peripheral Component Interconnect Express, высокоскоростная шина (x16 линий, до 64 ГТ/с).

Ход работы:

1. Создать client/server с socket (TCP localhost).
2. Сериализовать геоданные (numpy to bytes via tobytes()).
3. Передать пакеты 1KB-1MB, измерить bandwidth (time/bytes).

4. Тестировать UDP для lossy сценария.
5. Сравнить с theoretical max (Ethernet 125MB/s).
6. Графики bandwidth vs packet size.

Ожидаемый результат: Таблица bandwidth (TCP>90MB/s локально), графики, анализ overhead.

Контрольные вопросы:

1. Разница синхронных/асинхронных шин?
2. Как PCIe улучшает Ethernet?
3. Что ограничивает UDP throughput?

Лабораторная работа №6. Параллельная обработка геоданных

Цель работы: Реализовать параллелизм multiprocessing в Python, рассчитать ускорение по Амдалу для сейсмического моделирования.

Теоретический справочник:

Параллелизм — одновременное выполнение задач: битовый (в АЛУ), инструкционный (в конвейере), потоковый (гиперпотоки), задачный (процессы). SMP (Symmetric MultiProcessing) — равноправные процессоры с общей памятью; NUMA (Non-Uniform Memory Access) — память с разной скоростью доступа. Закон Амдала: $S(p) = 1 / (f + (1-f)/p)$, где f — доля последовательного кода, p — число процессоров. Multiprocessing — модуль Python для параллельных процессов (обходит GIL).

Ход работы:

1. Создать задачу: симуляция волны (FFT на геосетке numpy.fft).
2. Последовательная версия (1 процесс).
3. Параллельная (Pool 1-8 cores).
4. Рассчитать speedup, efficiency.
5. График $S(p)$ по Амдалу.
6. Тестировать на реальных CPU cores.

Ожидаемый результат: Графики speedup (linear до 4 cores), таблица Амдала, анализ для геомоделирования.

Контрольные вопросы:

1. Формула закона Амдала?
2. SMP vs NUMA?
3. Почему не идеальный speedup?

Лабораторная работа №7. Профилирование производительности

Цель работы: Профилировать Python-скрипт геообработки с cProfile, рассчитать MIPS/FLOPS и выявить bottlenecks для оценки систем.

Теоретический справочник:

MIPS — Million Instructions Per Second, миллионы инструкций в секунду. FLOPS — Floating Point Operations Per Second, операций с плавающей запятой в секунду. CPI — Cycles Per Instruction; IPC — Instructions Per Cycle ($1/CPI$). cProfile — встроенный профайлер Python, измеряет время вызовов функций. Bottleneck — узкое место, ограничивающее производительность (CPU, память, ИО). Linpack — бенчмарк для матричных вычислений (HPL — High Performance Linpack).[1][2][3]

Ход работы:

1. Написать скрипт: обработка геоданных (фильтр, FFT, stats).
2. Профилировать cProfile.run('script()', 'profile.prof').
3. Анализ: pstats (cumulative time, calls).
4. Рассчитать FLOPS (операции/time).
5. Сравнить варианты (numpy vs loops).
6. Выявить top bottlenecks.

Ожидаемый результат: Профиль-отчет (snakeviz), FLOPS (GFLOPS), рекомендации по оптимизации.

Контрольные вопросы:

1. Что измеряют MIPS и FLOPS в оценке производительности?
2. Как cProfile помогает выявить bottlenecks?
3. В чем разница CPI и IPC?

5.1.2 Методические указания для обучающихся по самостоятельной работе:

Рекомендации по самостоятельной работе:

1. Рекомендации по самостоятельной подготовке к лабораторным работам

- Изучите теоретический материал по теме лабораторной работы. Ознакомьтесь с учебниками, лекциями и дополнительными источниками, чтобы понимать цели и задачи работы, основные понятия и методы, используемые в лабораторном задании¹.
- Внимательно ознакомьтесь с методическими указаниями и требованиями к лабораторной работе. Обратите внимание на последовательность выполнения этапов, используемое программное обеспечение, форматы исходных и выходных данных, требования к визуализации и анализу результатов.
- Подготовьте исходные данные. Проверьте наличие всех необходимых файлов, убедитесь в их корректности (форматы, структура, отсутствие ошибок и пропусков данных).
- Освойте необходимые функции и инструменты программного обеспечения. Повторите работу с теми модулями и инструментами, которые будут использоваться в лабораторной работе.

• Планируйте время. Разделите выполнение работы на этапы: подготовка данных, выполнение анализа, оформление визуализации, написание отчета.

2. Рекомендации по оформлению отчетов по лабораторным работам

- Структурируйте отчет по стандартной схеме:
- Титульный лист (название работы, ФИО, группа, дата)
- Цель работы
- Краткое описание исходных данных
- Описание используемых методов и программного обеспечения
- Последовательное изложение этапов работы с иллюстрациями (скриншотами, графиками, картами)
- Анализ полученных результатов (выявленные особенности, сравнение с теорией, интерпретация)
- Выводы и рекомендации
- Список использованных источников
- Используйте качественные иллюстрации. Все графические материалы должны быть четкими, снабжены подписями, масштабами, легендами и пояснениями.
- Формулируйте выводы по существу. Кратко и ясно отражайте основные результаты работы, выявленные закономерности, достоинства и ограничения применяемых методов.
- Оформляйте отчет в соответствии с требованиями ДОТ. Соблюдайте стандарты оформления текста, таблиц, рисунков и ссылок на источники.

3. Рекомендации по самостоятельной проработке отдельных разделов тем

- Изучайте рекомендованную литературу и дополнительные источники. Используйте учебники, статьи, электронные ресурсы, профессиональные базы данных и справочные материалы, указанные в рабочей программе дисциплины¹.
- Выполняйте конспектирование ключевых понятий и алгоритмов. Составляйте краткие записи по основным определениям, алгоритмам, этапам работы с ПО, особенностям визуализации и анализа данных.

- Практикуйтесь в самостоятельном выполнении типовых заданий. Решайте задачи, связанные с обработкой и визуализацией геолого-геофизических данных, используя различные программные средства.
- Формулируйте вопросы и уточнения для обсуждения на занятиях. Записывайте непонятные моменты, чтобы получить разъяснения у преподавателя или в ходе дискуссии.
- Анализируйте примеры из практики. Изучайте реальные кейсы решения задач геофизики, сравнивайте разные подходы и делайте выводы о целесообразности их применения.

4. Общие рекомендации

- Развивайте навыки поиска и критического анализа информации. Пользуйтесь современными информационными ресурсами, анализируйте достоверность и актуальность найденных данных.
- Акцентируйте внимание на интеграции знаний и умений. Старайтесь связывать теоретические знания с практическими задачами, анализируйте, как выбранные методы и технологии влияют на качество и достоверность графического представления информации.
- Соблюдайте академическую честность. Все результаты, представленные в отчетах, должны быть получены самостоятельно, с обязательным указанием источников заимствованных данных и иллюстраций.

6 Фонд оценочных средств для контроля текущей успеваемости и проведения промежуточной аттестации по дисциплине

6.1 Оценочные средства для проведения текущего контроля

6.1.1 семестр 2 | Устный опрос

Описание процедуры.

Опрос может проводиться:

Фронтально — в форме беседы с группой, когда вопросы задаются всей группе, а ответы даются по очереди или по желанию.

Индивидуально — каждый студент отвечает на один или несколько вопросов, давая развернутый, связный ответ, часто с примерами и пояснениями.

Комбинированно — сочетаются оба подхода, а также используются дополнительные методы (например, письменные карточки, рецензирование ответов товарищей)

Критерии оценивания.

полнота и правильность ответа;
 понимание и осознанность материала;
 логичность и последовательность изложения;
 корректность терминологии;
 способность отвечать на уточняющие вопросы

6.2 Оценочные средства для проведения промежуточной аттестации

6.2.1 Критерии и средства (методы) оценивания индикаторов достижения компетенции в рамках промежуточной аттестации

Индикатор достижения	Критерии оценивания	Средства
----------------------	---------------------	----------

компетенции		(методы) оценивания промежуточной аттестации
ОПК ОС-7.1	полнота и правильность ответа; понимание и осознанность материала; логичность и последовательность изложения; корректность терминологии; способность отвечать на уточняющие вопросы	устный опрос

6.2.2 Типовые оценочные средства промежуточной аттестации

6.2.2.1 Семестр 2, Типовые оценочные средства для проведения зачета по дисциплине

6.2.2.1.1 Описание процедуры

К зачету допускаются студенты сдавшие все отчеты по лабораторным (практическим) работам. Зачёт проводится в форме устного опроса или тестирования, включающего 5 вопросов — по одному из каждой основной темы курса. В некоторых случаях допускается комбинированная форма: тест + устный опрос.

Время на ответ ограничено, ответы должны быть чёткими, логичными и аргументированными.

В случае неудовлетворительного результата студенту предоставляется возможность пересдачи в установленные сроки. При повторном не сдаче возможна дополнительная консультация и индивидуальное собеседование. Оценка выставляется по шкале с учётом полноты и правильности ответов.

Вопросы для подготовки к зачету:

1. Что такое классификация Флинна? Опишите SISD, SIMD, MISD, MIMD с примерами для геоданных.[1]
2. В чем разница принципов фон Неймана и Гарвардской архитектуры?
3. Как классификация по Флинну применяется к многоядерным системам в задачах недропользования?
4. Перечислите стадии конвейера процессора и их функции (fetch-decode-execute-writeback).
5. Что такое АЛУ? Какие операции оно выполняет?
6. Объясните понятия CPI и hazard в контексте процессора.
7. Опишите уровни иерархии памяти: L1, L2, L3, ОЗУ, виртуальная память.
8. Что такое локальность времени и пространства? Приведите примеры для геосеток.
9. Как работает политика LRU в кэше? Рассчитайте hit ratio для примера.
10. Различие блочных и символьных устройств ИО? Примеры для геофизических сенсоров.
11. Что такое DMA? Преимущества над PIO.
12. Опишите механизм прерываний: IRQ, вектор, приоритет.
13. Какие типы шин существуют (адресная, данных, управления)? Разница синхронных и асинхронных.
14. Сравните стандарты PCI/PCIe, USB, Ethernet по скорости и применению в геоданных.
15. Что такое CSMA/CD в Ethernet? Протоколы DMA over PCIe.
16. Уровни параллелизма: битовый, инструкционный, потоковый, задачный.

17. Разница SMP и NUMA? Применение в кластерах для сейсмообработки.
18. Формула закона Амдала? Рассчитайте speedup для $f=0.1$, $p=4$.
19. Определите MIPS, FLOPS, CPI, IPC. Как рассчитать FLOPS для геообработки?
20. Что такое bottlenecks? Методы профилирования (perf, cProfile).
21. Закон Мура и бенчмарки SPEC, Linpack для аппаратных комплексов ИС.
22. Принципы конфигурирования аппаратных комплексов для ГИС недропользования (CPU/GPU/память).
23. Как NumPy реализует SIMD-подобный параллелизм в лабораторной №1?
24. Рассчитайте CPI для конвейера с 2 hazards в лабораторной №2.
25. Почему row-major доступ быстрее column-major в кэше (лабораторная №3)?
26. Преимущества DMA+прерывания над CPU-polling в ИО (лабораторная №4).
27. Как измерить bandwidth сетевых интерфейсов socket в лабораторной №5?
28. Закон Густафсона vs Амдала: когда какой применять для геомоделирования.
29. Как cProfile выявляет bottlenecks? Разница MIPS и FLOPS (лабораторная №7).
30. Оцените совместимость компонентов для ИС недропользования: workstation vs сервер.
[2][3][1]

6.2.2.1.2 Критерии оценивания

Зачтено	Не зачтено
выставляется студенту, твердо знающему программный материал, грамотно и по существу его излагающему, который не допускает существенных неточностей в ответе на вопросы, правильно применяет теоретические положения при решении практических задач;	выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, не может ответить на дополнительные вопросы

7 Основная учебная литература

1. Архитектура ЭВМ и систем [Электронный ресурс] : методические указания к лабораторным работам и самостоятельной работе студентов / Иркут. гос. техн. ун-т, 2005. - 15.

[Сайт] – URL: <http://elib.istu.edu/viewer/view.php?file=/files/er-13767.pdf>

2. Игумнов И. В. Архитектура ЭВМ и систем : электронный курс / И. В. Игумнов, 2022

[Сайт] – URL: <https://el.istu.edu/course/view.php?id=6542>

8 Дополнительная учебная литература и справочная

1. Бройдо В. Л. Архитектура ЭВМ и систем : учебник для вузов по направлению подготовки дипломированных специалистов "Информационные системы" / В. Л. Бройдо, О. П. Ильина, 2009. - 720.
2. Новожилов О. П. Архитектура ЭВМ и систем : учебное пособие для студентов вузов по направлению 230100 "Информатика и вычислительная техника" / О. П. Новожилов, 2012. - 527.

9 Ресурсы сети Интернет

1. <http://library.istu.edu/>
2. <https://e.lanbook.com/>

10 Профессиональные базы данных

1. <http://new.fips.ru/>
2. <http://www1.fips.ru/>

11 Перечень информационных технологий, лицензионных и свободно распространяемых специализированных программных средств, информационных справочных систем

1. Лицензионное программное обеспечение Системное программное обеспечение
2. Лицензионное программное обеспечение Пакет прикладных офисных программ
3. Лицензионное программное обеспечение Интернет-браузер

12 Материально-техническое обеспечение дисциплины

1. Учебная аудитория для проведения лекционных занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Оснащение: комплект учебной мебели, рабочее место преподавателя, доска. Мультимедийное оборудование (в том числе переносное): мультимедийный проектор, экран, акустическая система, компьютер с выходом в интернет.
2. Учебная аудитория для проведения лабораторных/практических (семинарских) занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Оснащение: комплект учебной мебели, рабочее место преподавателя, доска. Мультимедийное оборудование (в том числе переносное): мультимедийный проектор, экран, акустическая система, компьютер с выходом в интернет.