

Документ подписан простой электронной подписью
 Информация о владельце:
 ФИО: АРЦИУХ ДМИТРИЙ ВЛАДИМИРОВИЧ
 Должность: Директор Организации
 Дата подписания: 01.05.2019
 Уникальный программный ключ:
 194e9de362a3e118beb1b1a74bc7c1577477a952

Оценочные материалы дополнительной общеразвивающей программы

«Искусственный интеллект в бизнесе для руководителей»

Промежуточная аттестация проводится в форме экзамена по модулю в виде выполнения практического задания по пройденным темам. Для успешного прохождения ПА слушатель выполняет по одной индивидуальной работе по каждому из модулей и размещает полный комплект материалов в ЛМС, согласно утверждённым критериям. Формат демонстрации не предусмотрен.

Критерии оценивания промежуточной аттестации:

- соответствие заданию модуля и полнота выполнения всех пунктов;
- корректность решений и валидность результатов;
- качество оформления;
- соблюдение академической добросовестности (оригинальность, корректные заимствования, отсутствие совпадений, не объяснённых ссылками);
- соблюдение сроков сдачи в ЛМС.

Шкалы оценивания:

Баллы/Критерии	соответствие заданию модуля и полнота выполнения всех пунктов	корректность решений и валидность результатов;	качество оформления;	соблюдение академической добросовестности	соблюдение сроков сдачи в ЛМС.
5	Выполнены все пункты задания модуля в полном объёме, нет пропусков, работа полностью соответствует заданию	Решения технически/логически верны, результаты воспроизводимы и обоснованы, обработка ошибок корректна	Структурировано, читаемо, единый стиль, код/текст отформатированы, есть комментарии/пояснения, файлы названы логично, материалы упакованы аккуратно	Работа полностью оригинальна (или все заимствования явно указаны ссылками), нет совпадений с другими работами, нет плагиата	--
4	Выполнены все основные пункты, но отсутствует 1 незначительный элемент / незначительная часть	Решения в целом верны, но есть 1–2 незначительные ошибки / неточности, не влияющие на основной результат	Хорошее оформление, но есть мелкие недочёты: местами нарушен стиль, не все файлы подписаны, отсутствуют отдельные комментарии	Единичные совпадения, объяснённые корректными ссылками; общая оригинальность высокая (>80%)	--
3	Выполнено 70–80% задания, отсутствуют 2–3 пункта или один значительный блок	Решения работают, но есть ошибки в краевых случаях / не обработаны исключения / результаты частично неверны	Среднее качество: структура есть, но оформление «грязное» (лишние файлы, разный стиль, мало комментариев)	Есть заимствования без ссылок (10–20% работы), или незначительные совпадения с другими слушателями	Работа сдана строго в установленный срок (включая все материалы)
2	Выполнено 50–69% задания, отсутствует более 3 пунктов	Решения содержат системные ошибки,	Оформление плохое: трудно читать, нет структуры,	Значительные заимствования без ссылок (20–40%), или	Незначительная задержка (до 3 календарных дней)

	или ключевая часть	результаты невалидны или не воспроизводятся	неясно, что к чему относится	заметные совпадения с другими работами	
1	Выполнено менее 50% задания, работа не соответствует заданию	Решения не работают, результаты отсутствуют или противоречивы	Оформление отсутствует (набор файлов без пояснений, нечитаемый код/текст)	Более 40% работы скопировано (без ссылок), или работа несамостоятельна	Задержка от 4 до 7 календарных дней
0	Работа не сдана или не относится к модулю	Решения не представлены или полностью неверны	Материалы не загружены в ЛМС или загружены с нарушением формата (не открываются)	Полный плагиат (работа целиком не своя, или копия другой работы)	Задержка более 7 календарных дней или работа не сдана

Итоговая шкала оценивания

Отлично	21–23 балла
Хорошо	17–20 баллов
Удовлетворительно	14–16 баллов
Неудовлетворительно	менее 14 баллов

Слушателям, получившие по результатам промежуточной аттестации оценку «неудовлетворительно» устанавливаются сроки повторной промежуточной аттестации. Если обучающийся не ликвидировал академическую задолженность при прохождении повторной промежуточной аттестации в первый раз, ему предоставляется возможность пройти повторную промежуточную аттестацию во второй раз.

Первая повторная промежуточная аттестация и вторая повторная промежуточная аттестация проводятся комиссией. Состав комиссии формируется департаментом образовательных программ.

Задание 1. Основы Linux и работа с файловой системой

В домашней директории создайте структуру каталогов и выполните операции с файлами согласно списку. Результат оформите в виде отчёта с командами и скриншотами.

Список действий:

Создать структуру: ~/exam/projects/, ~/exam/backups/, ~/exam/temp/

В projects создать файлы: readme.txt, notes.txt, todo.txt

Скопировать todo.txt в backups

Переместить notes.txt в temp

Установить на readme.txt права 640

Найти все .txt файлы в домашней директории

Эталон выполнения

Отчёт со всеми командами (например, mkdir, touch, cp, mv, chmod, find)

Скриншот вывода `ls -la ~/exam`

Скриншот вывода `find ~ -name "*.txt" 2>/dev/null`

Критерии оценки

Что проверяется	Балл
Все каталоги созданы	1
Все файлы созданы	1
Копирование и перемещение выполнены верно	1
Права 640 установлены	1
Поиск файлов выполнен	1
Итого	5

Задание 2. Bash-скрипт для резервного копирования

Напишите Bash-скрипт `backup.sh`, который принимает два аргумента: исходную директорию и директорию для резервного копирования.

Требования к скрипту:

Проверить, существует ли исходная директория. Если нет — вывести ошибку.

Создать директорию для резервного копирования, если её нет.

Скопировать все файлы с расширением `.txt` из исходной директории в целевую.

При копировании добавить к имени файла текущую дату в формате ГГГГ-ММ-ДД.

Вывести количество скопированных файлов.

Эталон выполнения

Текст скрипта (код)

Скриншот выполнения с примером

Скриншот результата (файлы с датой в имени)

Критерии оценки

Что проверяется	Балл
Проверка исходной директории	1
Создание целевой директории	1
Копирование только <code>.txt</code> файлов	1
Добавление даты к имени файла	1
Подсчёт и вывод количества файлов	1
Итого	5

Задание 3. Git: локальный и удалённый репозиторий

Выполните следующие действия с Git:

Настроить Git (имя и email пользователя)

Создать локальный репозиторий в папке `~/git-exam`

Создать файл `hello.txt` с текстом «Hello, Git!» и сделать первый коммит

Создать ветку `feature`, переключиться на неё

Добавить в `hello.txt` новую строку «Feature branch work» и сделать коммит

Переключиться обратно в ветку `main`

Создать репозиторий на GitHub

Связать локальный репозиторий с удалённым
Отправить ветку main на GitHub

Эталон выполнения

Отчёт со всеми командами

Скриншот вывода `git log --oneline`

Скриншот репозитория на GitHub с коммитом

Ссылка на репозиторий (или скриншот)

Критерии оценки

Что проверяется	Балл
Настройка Git выполнена	1
Локальный репозиторий и первый коммит	1
Ветка feature и коммит в ней	1
Удалённый репозиторий создан и связан	1
Код отправлен (push) на GitHub	1
Итого	5

Задание 4. Программа на C: «Угадай число»

Напишите программу на C, которая реализует игру «Угадай число».

Требования:

Программа загадывает случайное число от 1 до 100

Пользователь вводит числа, программа даёт подсказки: «Больше», «Меньше», «Угадал!»

У пользователя не более 10 попыток

После окончания игры программа спрашивает: «Сыграем ещё? y/n»

При вводе y — новая игра, при n — выход

Эталон выполнения

Исходный код программы (файл guess.c)

Скриншот компиляции (`gcc guess.c -o guess`)

Скриншот выполнения программы (с примером игры)

Критерии оценки

Что проверяется	Балл
Программа компилируется без ошибок	1
Случайное число генерируется	1
Подсказки «Больше»/«Меньше» работают	1
Ограничение на 10 попыток	1
Возможность сыграть ещё	1
Итого	5

Задание 5. Массивы и функции на C

Напишите программу на C, которая работает с массивом целых чисел.

Требования:

Запросить у пользователя размер массива N (не более 20)

Заполнить массив числами, введёнными с клавиатуры

Вывести исходный массив на экран

Найти и вывести минимальный, максимальный элементы и сумму всех элементов (каждое вычисление в отдельной функции)

Отсортировать массив по возрастанию методом «пузырька»

Вывести отсортированный массив

Подсчитать и вывести количество чётных чисел

Эталон выполнения

Исходный код программы (файл array.c)

Скриншот компиляции и выполнения (например, для N=5)

Критерии оценки

Что проверяется	Балл
Ввод N и заполнение массива	1
Функции минимума, максимума, суммы	1
Сортировка «пузырьком»	1
Вывод исходного и отсортированного массивов	1
Подсчёт чётных чисел	1
Итого	5

Задание 6. Работа с файлами на C

Напишите программу file_stats.c, которая анализирует текстовый файл.

Требования:

Программа принимает аргументы командной строки: входной файл и выходной файл

Проверяет, что передано 2 аргумента, иначе выводит инструкцию

Открывает входной файл для чтения, если файл не существует — выводит ошибку

Подсчитывает во входном файле:

количество строк

количество слов (разделители: пробел, табуляция, перевод строки)

количество символов (без учёта перевода строки)

Записывает результаты в выходной файл в формате:

text

Количество строк: X

Количество слов: Y

Количество символов: Z

Закрывает оба файла

Эталон выполнения

Исходный код программы (файл file_stats.c)

Скриншот компиляции
Пример файла input.txt
Скриншот выполнения программы
Полученный файл output.txt (содержимое)
Критерии оценки

Что проверяется	Балл
Проверка аргументов командной строки	1
Проверка существования входного файла	1
Корректный подсчёт строк, слов, символов	1
Запись результатов в выходной файл	1
Программа компилируется и работает	1
Итого	5