

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Сурат Лев Игоревич
Должность: Ректор
Дата подписания: 11.08.2026 18:33:05
Уникальный программный ключ:
90e61d348f2245de4566514a87350a9d89d73c851b3f3160a03a9eff20fb4800

**Московский
Институт
Психоанализа**

Негосударственное образовательное частное учреждение
высшего образования «Московский институт психоанализа»
(НОЧУ ВО «Московский институт психоанализа»)

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ОСНОВЫ ПРЕПОДАВАНИЯ ИНФОРМАТИКИ

Направление подготовки/специальность	44.03.05 Педагогическое образование (с двумя профилями подготовки)
Направленность (профиль/специализация)	Иностранный язык (английский) и информационные технологии в образовании
Уровень высшего образования	Бакалавриат
Форма обучения	Очная

1. Фонд оценочных средств для проведения текущего контроля успеваемости

Задания для проведения текущего контроля успеваемости (темы докладов, рефератов, эссе, тестовые задания, кейсовые задания и т.п.) (а также, обязательно представить задания для подготовки к практическим занятиям, а также критерии и шкалу оценивания к ним)

Вопросы для устного опроса.

1. Раскройте сущность понятия «алгоритмическое мышление» (*Computational Thinking*). Как соотносятся декомпозиция, абстракция и распознавание паттернов в структуре учебной деятельности школьника?
2. Психолого-педагогические особенности усвоения темы «Представление информации». Трудности перехода от непрерывных величин к дискретным (проблема квантования).
3. Концепция многоуровневой модели памяти при изучении программирования (Рабочая память vs Долговременная память). Когнитивная нагрузка *Intrinsic/Extraneous/Germane* на уроках кодирования.
4. Формирование навыка отладки кода (*Debugging*) как метакогнитивный процесс. Методология обучения поиску логических ошибок без использования ИИ-ассистентов.
5. Дидактическая единица «Исполнитель алгоритма». Анализ ограничений формальных систем через примеры машин Тьюринга и нормальных алгоритмов Маркова.
6. Проблема формирования понятий «Переменная» и «Тип данных». Семантический разрыв между математической переменной и ячейкой памяти в представлении ученика.
7. Системный подход к изучению архитектуры ЭВМ (Архитектура фон Неймана). Принципы взаимодействия аппаратной и программной компонент («Hard–Soft interface»).
8. Обучение информационной безопасности (*Cyber Hygiene*) в начальной школе. Адаптация сложных криптографических концепций под возрастные особенности восприятия.
9. Визуальное программирование (*Scratch, Blockly*) versus текстовое кодирование. Критерии готовности обучающегося к переходу от блочных структур к синтаксису языков высокого уровня.
10. Этические аспекты внедрения искусственного интеллекта в образование. Проблема академической честности при генерации программного кода нейросетями.
11. Методика изучения электронных таблиц не как калькулятора, а как среды для моделирования (*What-if analysis*). Использование циклических ссылок и итерационных вычислений.
12. Технология обучения базам данных. Проектирование реляционной схемы (*ER-диаграммы*) учениками: переход от предметной области к нормализации таблиц.
13. Робототехника (*LEGO Mindstorms, Arduino*) как средство материализации абстрактных алгоритмов. Синхронизация программной логики и физических датчиков.
14. Компьютерное моделирование (*Имитационное моделирование*). Различие между детерминированными и стохастическими моделями в учебных проектах.
15. Методика преподавания сетевых технологий и протоколов (*OSI/TCP-IP model*). Визуализация пакетной коммутации данных для школьников.
16. Мультимедиа-дидактика. Создание образовательного контента: баланс между когнитивной теорией мультимедийного обучения (Майер) и техническими возможностями ПО.

17. Организация исследовательской деятельности учащихся средствами языка Python (библиотеки matplotlib, pandas). Интерпретация больших данных (*Big Data literacy*).
18. Метод проектов в информатике. Оценка IT-проекта: критерии работоспособности кода, качества UI/UX и документирования (README-файлы).
19. Дифференцированный подход при обучении решению задач ОГЭ/ЕГЭ по информатике. Работа со сложностью алгоритмических конструкций (вложенные циклы, рекурсия).
20. Формирующее оценивание навыков программирования. Роль рубрикаторов (*Rubrics*) при проверке псевдокода и блок-схем до этапа компиляции.
21. Санитарно-гигиенические нормы работы за компьютером (СанПиН). Профилактика компьютерного зрительного синдрома и нарушений осанки у обучающихся.
22. Цифровая трансформация педагога. Переход от роли транслятора знаний к роли архитектора образовательной среды (*Learning Designer*) в LMS.
23. Свободное программное обеспечение (*Open Source*) в российской школе. Лицензионная чистота используемого ПО и правовые риски.
24. Инклюзивное обучение информатике. Адаптация интерфейсов и методик ввода текста для детей с нарушениями опорно-двигательного аппарата (НОДА).
25. Лабораторный практикум по информатике. Техника безопасности при работе с периферийным оборудованием и электроприборами в кабинете ИКТ.

Критерии и шкалы оценивания работы студентов на практических/лабораторных занятиях

Оценка	Критерии оценивания
«отлично»	Оценка «отлично» выставляется, если студент активно работает в течение всего практического занятия, дает полные ответы на вопросы преподавателя в соответствии с планом практического занятия и показывает при этом глубокое овладение лекционным материалом, знание соответствующей литературы и законодательства, способен выразить собственное отношение по данной проблеме, проявляет умение самостоятельно и аргументированно излагать материал, анализировать явления и факты, делать самостоятельные обобщения и выводы, правильно выполняет учебные задачи.
«хорошо»	Оценка «хорошо» выставляется при условии: студент активно работает в течение практического занятия, вопросы освещены полно, изложения материала логическое, обоснованное фактами, со ссылками на соответствующие нормативные документы и литературные источники, освещение вопросов завершено выводами, студент обнаружил умение анализировать факты и события, а также выполнять учебные задания. Но в ответах допущены неточности, некоторые незначительные ошибки, имеет место недостаточная аргументированность при изложении материала, четко выраженное отношение студента к фактам и событиям.
«удовлетворительно»	Оценка «удовлетворительно» выставляется в том случае, когда студент в целом овладел сути вопросов по данной теме, обнаруживает знание лекционного материала, законодательства и учебной литературы, пытается анализировать факты и события, делать выводы и решать задачи. Но на занятии ведет себя пассивно, отвечает только по вызову преподавателя, дает неполные ответы на вопросы, допускает грубые ошибки при освещении теоретического материала.
«неудовлетворительно»	Оценка «неудовлетворительно» выставляется в случае, когда студент обнаружил несостоятельность осветить вопросы освещены неправильно, бессистемно, с грубыми ошибками, отсутствуют понимания основной сути вопросов, выводы, обобщения.

2. Фонд оценочных средств для проведения промежуточной аттестации

Задания для проведения промежуточной аттестации по итогам освоения дисциплины (тестовые задания, кейсовые задания и т.п.)

1. Согласно концепции Cognitive Load Theory (Свелер), эффективная методика обучения программированию должна минимизировать нагрузку типа:
А) Germane (сущностную).
Б) Extraneous (постороннюю/избыточную), вызванную неудачным дизайном интерфейса или запутанным объяснением.
В) Intrinsic (внутреннюю), так как она неизменна.
Г) Физическую нагрузку на пальцы.
2. Основное дидактическое преимущество визуального программирования (Block-based coding) перед текстовым заключается в:
А) Возможности создания более быстрых программ.
Б) Устранении барьера синтаксического оверхеда (*Syntax overhead*), позволяющем сосредоточиться на логике алгоритма.
В) Отсутствии необходимости использовать мышь.
Г) Большем объеме занимаемой оперативной памяти.
3. При изучении темы «Системы счисления» типичной ошибкой является путаница между основанием системы и:
А) Количеством используемых цифр (мощность алфавита), где основание n использует цифры от 000 до $n-1$.
Б) Цветом чернил в тетради.
В) Временем суток.
Г) Номером страницы учебника.
4. В контексте базы данных термин "Нормализация" означает процесс:
А) Увеличения размера файла базы данных.
Б) Минимизации избыточности данных и исключения аномалий вставки/модификации путем декомпозиции отношений (таблиц).
В) Удаления всех записей из таблицы.
Г) Изменения шрифта в SQL-запросе.
5. Принцип локальности ссылок (*Locality of reference*) в архитектуре ЭВМ объясняет эффективность использования:
А) Иерархии кэш-памяти (данные, использованные недавно, скорее всего, будут использованы снова).
Б) Большого количества USB-портов.
В) Яркого освещения в классе.
Г) Скорости печати принтера.
6. Методика Parson's Puzzles используется в обучении программированию для развития навыка:
А) Рисования графиков.
Б) Анализа структуры кода и понимания правил композиции операторов без необходимости их набора с нуля.
В) Игры в шахматы.
Г) Быстрого выключения компьютера.
7. Ключевое отличие имитационного моделирования от аналитического заключается в том, что оно:
А) Использует только бумагу и карандаш.
Б) Воспроизводит функционирование системы во времени через пошаговое выполнение алгоритма состояний (State transitions).
В) Всегда дает точный результат без погрешностей.
Г) Не требует знания математики.

8. В робототехнике понятие "Dead Reckoning" (инерциальная навигация) описывает метод позиционирования, основанный на:
 - А) Сигналах GPS спутника.
 - Б) Интеграции показаний энкодеров колес относительно известной стартовой точки (накопление ошибки позиционирования).
 - В) Подсказках учителя.
 - Г) Магнитном поле Земли.
9. Библиотека Matplotlib в языке Python предназначена для:
 - А) Управления базой данных.
 - Б) Статической, интерактивной и анимационной визуализации данных (Data Visualization).
 - В) Отправки электронной почты.
 - Г) Архивации файлов.
10. Феномен "Notional Machine" (Абстрактная машина) в дидактике информатики — это:
 - А) Реальный суперкомпьютер в лаборатории.
 - Б) Когнитивная модель того, как работает компьютер, которой пользуется программист для предсказания поведения кода.
 - В) Учебник по физике.
 - Г) Название антивируса.
11. Использование глобальных переменных (*Global variables*) в учебном коде считается плохой практикой, так как это нарушает принцип:
 - А) Экономии места на диске.
 - Б) Инкапсуляции и минимизации побочных эффектов (*Side effects*), затрудняя отладку.
 - В) Красоты оформления.
 - Г) Скорости интернета.
12. Протокол HTTPS отличается от HTTP наличием слоя:
 - А) Шифрования трафика (TLS/SSL) для обеспечения конфиденциальности и целостности передаваемых данных.
 - Б) Дополнительных картинок.
 - В) Звукового сопровождения.
 - Г) Другого цвета фона сайта.
13. Дидактическая единица "Алгоритм Евклида" чаще всего используется для демонстрации принципа:
 - А) Рекурсии или итерации нахождения НОД (Наибольшего общего делителя).
 - Б) Умножения матриц.
 - В) Работы поисковых систем.
 - Г) Строительства мостов.
14. Когнитивный феномен "Variable Assignment Misconception" (Заблуждение присваивания) проявляется, когда ученик думает, что оператор $x = x + 1$:
 - А) Является математическим уравнением без решений.
 - Б) Невозможен, вместо понимания его как команды обновления состояния ячейки памяти (Imperative paradigm).
 - В) Приводит к взрыву компьютера.
 - Г) Увеличивает размер экрана.
15. Технология "Pair Programming" (Парное программирование) направлена на улучшение:
 - А) Скорости печати двумя руками.
 - Б) Качества кода через механизмы немедленного обзора (*Real-time review*) и разделения ролей (Driver/Navigator).
 - В) Количества компьютеров в классе.

- Г) Громкости обсуждения.
16. В курсе школьной информатики электронная таблица рассматривается как инструмент:
- А) Только для рисования диаграмм.
 - Б) Табличного представления данных и реализации вычислительных моделей ("Что-если" анализ).
 - В) Замены бумажного дневника.
 - Г) Просмотра видео.
17. Критерий эффективности алгоритма сортировки обычно оценивается через:
- А) Количество строк кода.
 - Б) Временную сложность в нотации "О большое" ($O(n^2)$ vs $O(n \log n)$) в худшем случае.
 - В) Имя автора алгоритма.
 - Г) Цвет интерфейса программы.
18. Обработка исключений (*Exception Handling / Try-Except*) вводится в обучение для формирования навыка:
- А) Написания бесконечных циклов.
 - Б) Создания отказоустойчивых программ, корректно реагирующих на непредвиденные входные данные или сбои среды выполнения.
 - В) Удаления операционной системы.
 - Г) Игнорирования ошибок пользователя.
19. Понятие "Boolean Algebra" (Булева алгебра) является фундаментом для проектирования:
- А) Интерьера кабинета.
 - Б) Логических вентилей процессора и условных конструкций в языках программирования.
 - В) Правил дорожного движения.
 - Г) Расписания звонков.
20. Формирование цифровой грамотности (*Digital Literacy*), согласно ЮНЕСКО, включает умение:
- А) Только включать монитор.
 - Б) Находить, оценивать, создавать информацию и обмениваться ею с использованием цифровых технологий (Information fluency).
 - В) Только играть в игры.
 - Г) Чистить клавиатуру.
21. Методика "Faded Worked Examples" (Затухающие примеры) предполагает постепенное удаление фрагментов готового решения с целью:
- А) Запутать ученика.
 - Б) Поэтапной передачи когнитивного контроля от примера к ученику (Cognitive apprenticeship).
 - В) Экономии тонера принтера.
 - Г) Увеличения объема домашнего задания.
22. Рекурсия в программировании характеризуется обязательным наличием двух компонентов:
- А) Начала и конца учебного года.
 - Б) Базового случая (*Base case*) для остановки и рекурсивного шага, приближающего к нему.
 - В) Монитора и мыши.
 - Г) Учителя и директора.
23. Стандарт IEEE 829 регламентирует документацию процесса:
- А) Приготовления обеда.
 - Б) Тестирования программного обеспечения (Test Documentation Standard).

- В) Посадки деревьев.
 Г) Написания стихов.
24. Концепция "Unplugged Activities" (Активности без компьютера) в преподавании информатики доказывает, что суть СТ (Computational Thinking):
 А) Зависит от бренда процессора.
 Б) Носит фундаментальный характер и может быть освоена через физические манипуляции без использования ПК.
 В) Требуется наличия Wi-Fi.
 Г) Понятна только взрослым.
25. Основной риск использования облачных IDE (например, GitHub Codespaces) в школьном образовании:
 А) Отсутствие электричества в розетке.
 Б) Проблемы обработки персональных данных несовершеннолетних на зарубежных серверах (Compliance with data protection laws like FZ-152).
 В) Слишком яркие цвета редактора.
 Г) Наличие бесплатного тарифа.

Ключи к тестовым заданиям: 1-Б, 2-Б, 3-А, 4-Б, 5-А, 6-Б, 7-Б, 8-Б, 9-Б, 10-Б, 11-Б, 12-А, 13-А, 14-Б, 15-Б, 16-Б, 17-Б, 18-Б, 19-Б, 20-Б, 21-Б, 22-Б, 23-Б, 24-Б, 25-Б.

Критерии и шкала оценивания:

Оценка	Количество баллов
«отлично» /зачтено	83-100 баллов
«хорошо»/ зачтено	68-82 балла
«удовлетворительно»/ зачтено	50-67 баллов
«неудовлетворительно»/ не зачтено	0-49 баллов