

Педагогическая производственная практика

Инвариантное задание №7

Выполнила: студент группы «10м_КЭО» Леонтьева Анна Викторовна

Самоанализ профессиональной деятельности в рамках педагогической практики

Контекст практики:

В рамках прохождения педагогической практики мною были проведены два урока информатики для учащихся 11 класса по теме "Энергия связи атомных ядер". Целью данных уроков было не только освоение физических концепций, но и развитие практических навыков работы с табличными процессорами (Microsoft Excel) и основами программирования на языке Python для решения прикладных задач.

Тема уроков: "Энергия связи атомных ядер: табличный метод расчета и создание калькулятора на Python"

1. Цели уроков

Образовательные:

- Познакомить учащихся с фундаментальными понятиями ядерной физики: энергия связи ядра, дефект массы, удельная энергия связи ядра.
- Освоить формулы для расчета указанных величин.
- Сформировать навыки применения табличных процессоров (Excel) для выполнения сложных расчетов и анализа данных.
- Сформировать базовые навыки создания простых программ на Python для автоматизации вычислений и взаимодействия с пользователем.

Развивающие:

- Развивать логическое мышление, аналитические способности и навыки работы с информацией.
- Развивать умение переводить физические задачи в алгоритмическую форму и реализовывать их с помощью программных средств.
- Развивать навыки самостоятельной работы, поиска и устранения ошибок (отладки).
- Развивать межпредметные связи между физикой и информатикой.

Воспитательные:

- Воспитывать аккуратность и точность при работе с данными и программным кодом.
- Формировать интерес к изучению информатики и естественных наук через практическое применение знаний.

2. Содержание и структура уроков:

Оба урока были построены как практико-ориентированные занятия, где теоретические основы немедленно применялись для решения конкретных задач с использованием информационных технологий.

Урок 1: Табличный метод расчета в Excel

- Введение в тему (10-15 мин): Краткий теоретический блок, охватывающий определения энергии связи, дефекта массы, удельной энергии связи. Были представлены ключевые формулы: $M_{\text{я}} = M_{\text{а}} - Z \cdot m_{\text{е}}$, $\Delta m = (Z \cdot m_{\text{р}} + N \cdot m_{\text{н}}) - M_{\text{я}}$, $E_{\text{св}} = \Delta m \cdot c^2$, $E_{\text{уд}} = E_{\text{св}} / A$. Визуальное сопровождение (презентация с изображениями ядер, нуклонов, формул).
- Практическая работа в Excel (25-30 мин):
 - Шаг 1-4: Совместное создание структуры таблицы в Excel (заголовки, единицы измерения). Учащиеся копировали начальные данные (элемент, массовое число, атомный номер, масса атома) из предоставленных справочных материалов.
 - Шаг 5: Вычисление числа нейтронов ($N = A - Z$) с демонстрацией автозаполнения формул.
 - Шаг 6: Копирование констант (масса протона, нейтрона, электрона, коэффициент перевода, скорость света) в отдельный блок для удобства использования. Особое внимание было уделено использованию абсолютных ссылок (\$) для констант.
 - Шаг 7-12: Пошаговое вычисление остальных величин: массы ядра ($M_{\text{я}}$), дефекта массы в а.е.м. и кг (Δm), энергии связи в Дж и МэВ ($E_{\text{св}}$), удельной энергии связи в Дж и МэВ ($E_{\text{уд}}$). На каждом этапе демонстрировалось применение соответствующих формул Excel и автозаполнение. Была подчеркнута важность форматирования ячеек для корректного отображения экспоненциальных чисел.
 - Шаг 13: Финальное форматирование таблицы для наглядности.
- Рефлексия (5 мин): Обсуждение полученных результатов, их физического смысла, преимуществ табличного метода для большого объема данных.

Урок 2: Написание калькулятора на Python

- Повторение и постановка задачи (10 мин): Краткое повторение физических концепций и постановка задачи: создание интерактивного калькулятора на Python, который позволит рассчитывать те же величины для выбранного элемента. Обсуждение ожидаемого вывода программы.
- Разработка программы на Python (30-35 мин):

- Часть 1: Физические константы: Объявление констант в Python (например, `MP_AEM`). Обсуждение соглашений именования констант (заглавные буквы) и роли комментариев.
- Часть 2: Данные элементов: Создание словаря `elements_data` для хранения свойств элементов. Объяснение структуры "словарь в словаре" и удобства такого подхода для организации данных.
- Часть 3: Функция для расчетов (`calculate_nuclear_properties`):
 - Определение функции, принимающей название элемента и его данные.
 - Использование метода `.get()` для безопасного извлечения данных из словаря.
 Объяснение, почему это предпочтительнее прямого обращения по ключу (обработка отсутствующих ключей).
 - Реализация всех расчетных формул, аналогично Excel, но уже в синтаксисе Python.
 - Важный момент: проверка `if A > 0`: перед делением для удельной энергии связи, чтобы избежать `ZeroDivisionError`.
 - Возврат результатов в виде словаря для удобного доступа к вычисленным значениям.
- Часть 4: Основная часть программы (`main` функция):
 - Инициализация функции `main()`.
 - Вывод приветствия и списка доступных элементов. Использование `" ".join()` для красивого вывода списка.
 - Реализация бесконечного цикла `while True`: для многократного взаимодействия с пользователем, прерываемого вводом 'выход'.
 - Обработка пользовательского ввода: `.strip()` для удаления пробелов, `.lower()` для независимости от регистра при сравнении с 'выход'. Обсуждение, почему `.lower()` не применяется при проверке `if user_input in elements_data`.
 - Вызов функции `calculate_nuclear_properties` и вывод отформатированных результатов. Использование f-строк и спецификаторов форматирования (`:.6e`, `:.6f`) для численного вывода.
 - Обработка случая, когда элемент не найден.
- Часть 5: Точка входа в программу (`if __name__ == "__main__":`): Объяснение стандартной конструкции для запуска Python-скрипта.
- Рефлексия (5 мин): Сравнение двух подходов (Excel vs. Python), обсуждение, когда какой инструмент более эффективен. Демонстрация работы готового калькулятора.

3. Анализ проведенных уроков

Сильные стороны:

- **Межпредметность:** Уроки успешно продемонстрировали практическое применение знаний по физике в рамках курса информатики, что значительно повысило интерес учащихся и показало им реальную ценность IT-инструментов.
- **Практико-ориентированный подход:** Учащиеся не просто слушали теорию, но активно создавали инструменты для решения задач, что способствовало глубокому пониманию материала и развитию прикладных навыков.
- **Поэтапное обучение:** Пошаговая инструкция с подробными объяснениями каждого этапа (как в Excel, так и в Python) позволила большинству учащихся успешно справиться с заданиями.
- **Разнообразие инструментов:** Использование двух разных инструментов (Excel и Python) для решения одной и той же задачи позволило учащимся увидеть разные подходы к вычислениям и оценить их преимущества и недостатки.
- **Актуальность:** Расчеты в Excel и написание программ на Python – это востребованные навыки в современном мире.
- **Визуализация:** Презентационные материалы с формулами и скриншотами Excel/Python кода помогали учащимся ориентироваться и следовать инструкциям.

Аспекты для улучшения

- **Управление временем и темпом:** Тема достаточно объемна и включает в себя множество технических деталей. Несмотря на пошаговый подход, некоторые учащиеся могли отставать или не успевать освоить все тонкости за отведенное время. В будущем стоит рассмотреть возможность выделения большего количества часов на данную тему или разделения её на более мелкие, менее плотные блоки.
- **Дифференциация обучения:** Уровень подготовки учащихся по информатике и физике может существенно различаться. На уроках не в полной мере была реализована дифференциация:
 - Для более сильных учащихся можно было предложить дополнительные задачи или самостоятельное исследование других элементов.
 - Для учащихся с затруднениями стоило бы предусмотреть готовые шаблоны в Excel или частично написанный код в Python, чтобы они могли сосредоточиться на ключевых моментах, не отвлекаясь на базовый синтаксис.
- **Акцент на отладку:** Хотя была возможность задавать вопросы, не всегда хватало времени для полноценного обучения навыкам самостоятельной отладки формул в Excel и

кода в Python. В дальнейшем стоит включить небольшие "провокации" с ошибками, чтобы учащиеся учились их находить и исправлять.

- Обратная связь и оценка: В рамках практики оценка носила преимущественно формирующий характер (наблюдение за работой, проверка промежуточных результатов). Для более объективной оценки можно было бы предусмотреть короткий тест или задание на дом, где учащиеся могли бы применить полученные навыки без прямой помощи.
- Предварительная проверка знаний: В начале занятий полезно было бы провести короткий опрос или тест для определения уровня владения Excel и Python, чтобы скорректировать темп и глубину изложения.

4. Взаимодействие с учащимися и их вовлеченность:

Учащиеся 11 класса проявили значительный интерес к теме, особенно к практической части. Возможность "потрогать" физику через информатику, увидеть, как формулы оживают в электронных таблицах и программах, стимулировала их любопытство.

Большинство активно работали, задавали вопросы по ходу выполнения заданий, демонстрировали готовность к освоению нового материала. Наблюдалось моменты взаимопомощи между учащимися, что является положительным показателем.

Затруднения чаще всего возникали при точном следовании синтаксису формул в Excel (например, отсутствие скобок, неправильное использование \$) и при написании Python кода (ошибки в отступах, опечатки в названиях переменных).

5. Выводы и перспективы:

Проведенные уроки стали ценным опытом в моей педагогической практике. Они подтвердили эффективность межпредметного подхода и важность практических заданий для формирования устойчивых навыков. Я укрепился во мнении, что интеграция информационных технологий в преподавание естественных наук позволяет не только углубить понимание предмета, но и развить критически важные для современного мира компетенции.

В дальнейшем я планирую:

- Более тщательно планировать темп урока, учитывая сложность материала и уровень подготовки учащихся.
- Разрабатывать дифференцированные задания для учащихся с разным уровнем подготовки.

- Уделять больше внимания формированию навыков самостоятельной отладки и поиска решений.
- Активнее использовать элементы групповой работы для стимулирования взаимообучения.

В целом, считаю, что поставленные цели уроков были достигнуты, а учащиеся получили ценный опыт работы с реальными данными и современными вычислительными инструментами.