

2. *Курицына, Г. В.* Формы и методы контроля качества дистанционного обучения студентов вуза / Г. В. Курицына. Текст: электронный // Международный журнал экспериментального образования. 2014. № 8-3. С. 17–21. URL: <https://expeducation.ru/ru/article/view?id=5922>.

3. *Подласый, И. П.* Педагогика: 100 вопросов – 100 ответов / И. П. Подласый. Москва: ВЛАДОС-ПРЕСС, 2006. 365 с. Текст: непосредственный.

4. *Сергеев, И. С.* Компетентностный подход в обучении юридическим дисциплинам / И. С. Сергеев. Санкт-Петербург: Питер, 2014. 126 с. Текст: непосредственный.

5. *Симонов, В. П.* Диагностика степени обученности обучающихся / В. П. Симонов. Москва: МПА, 1999. 46 с. Текст: непосредственный.

УДК 378.147.88:004.738

Штанюк А. А.

РАЗРАБОТКА ИНТЕРАКТИВНЫХ УЧЕБНЫХ СИСТЕМ С ИСПОЛЬЗОВАНИЕМ JUPYTER NOTEBOOK, GITHUB И BINDER

Антон Александрович Штанюк

Кандидат технических наук, доцент

ashtanyuk@nntu.ru

Нижегородский государственный технический университет

им. Р. Е. Алексеева

DEVELOPING INTERACTIVE LEARNING SYSTEMS USING JUPYTER NOTEBOOK, GITHUB AND BINDER

Anton Shtanyuk

Nizhny Novgorod State Technical University, Russia, Nizhny Novgorod

Аннотация. Работа анализирует возможности создания интерактивных учебных систем при обучении информатике и программированию с использованием платформы GitHub, веб-среды Jupyter Notebook и технологии

Binder, основанной на контейнерной виртуализации. Данные технологии позволяют разрабатывать материал для практических занятий с интерактивными возможностями через сеть Интернет.

Abstract. *The paper analyses the possibilities for creating interactive learning systems for computer science and programming courses using the GitHub platform, the Jupyter Notebook web environment and Binder technology based on container virtualization.*

Ключевые слова: *Jupyter Notebook, GitHub, Binder, учебный процесс, программный код, программирование.*

Keywords: *Jupyter Notebook, GitHub, Binder, education process, programming.*

В процессе подготовки студентов информационных специальностей ВУЗов большое внимание уделяется современным технологиям работы с учебным материалом. Довольно популярными и распространёнными инструментами стали многочисленные онлайн-ресурсы, на которых преподаватели размещают задания, принимают решения от студентов, выполняют автоматические проверки этих решений [1].

Во многих вузах развернуты системы поддержки учебного процесса (в том числе основанные на широко известной Moodle), благодаря которым удастся решить множество проблем: от публикации учебных материалов до проведения тестирования [2]. При этом основной методикой работы с учебным кодом является его копирование на локальные компьютеры и выполнение под управлением локальных инструментов разработчика. К сожалению, система Moodle не всегда удобна для курсов, где используются большие примеры программного кода и есть необходимость выполнять учебный код, изменяя исходные данные, а также вносить исправления непосредственно в данный код. Все больше внимания привлекают системы с поддержкой интерактивного подхода, позволяющие студентам вносить исправления в учебный код и экспериментировать.

Из подобных систем наибольшее распространение получил Jupyter Notebook, — интерактивная веб-среда для разработки программ, поддерживающая текстовые блоки с разметкой на основе Markdown и Html, вставку изображений, являющихся результатом выполнения кода (например, графики и диаграммы) [3]. Изначально данное приложение поддерживало язык Python с его многочисленными библиотеками, что обусловило широкое распространение среди технических специалистов в области машинного обучения и обработки данных. В настоящее время поддерживаются и другие языки программирования (C++, C# и ряд других), что позволяет использовать данный инструмент в современных курсах программирования. Для преподавателя Jupyter оказался очень полезным инструментом [4, 5], позволяющим эффективно решать следующие задачи:

1. Подготовка качественно оформленных учебных материалов, насыщенных формулами, программным кодом и графиками, в том числе для конспектов и презентаций.
2. Чередование блоков с описаниями/заданиями и с реальным кодом, который можно выполнить на сервере и получить результаты прямо в текущем документе, то есть проведение вычислительных экспериментов «на месте».
3. Визуализировать результаты выполнения программ, благодаря программным средствам Python.

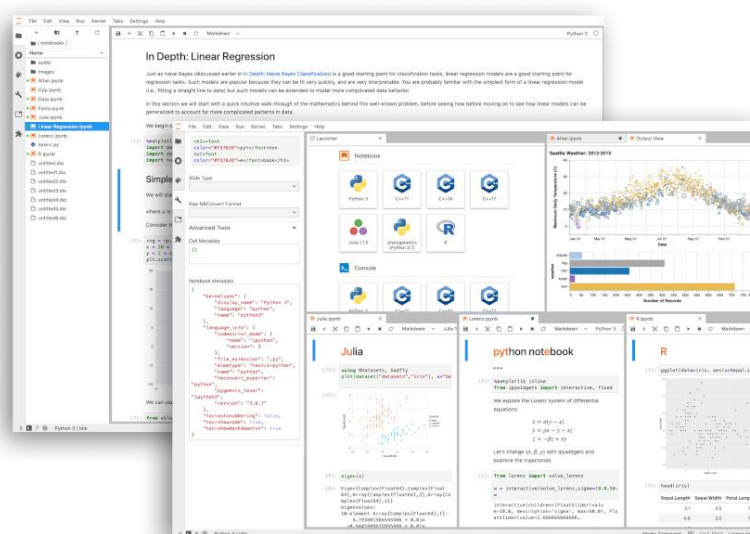


Рисунок 1 — Jupyter Notebook

Jupyter Notebook является свободно-распространяемым ПО, которое можно скачать с официального сайта и установить на локальные компьютеры или внутреннюю сеть учебного заведения. Но в этом случае могут возникнуть сложности, связанные с администрированием и поддержкой, что делает интернет-версию Jupyter более предпочтительной в ряде случаев.

Несколько лет назад компания Google запустила облачную версию, использующую Jupyter Notebook под названием Google Colab [6], что позволило перенести лабораторные занятия непосредственно «в облака», используя вычислительные мощности компании Google. Дополнительным плюсом является поддержка версией русского языка и наличие уже готовых учебников и пособий по машинному обучению и искусственному интеллекту.

В оставшейся части нашей работы мы рассмотрим использование интерактивных версий Jupyter Notebooks в репозиториях GitHub, который является крупнейшим хостингом для совместной разработки [7]. В последние годы GitHub активно осваивается как средство поддержки учебного процесса. На GitHub организуется совместная работа студентов над программным кодом, тестирование и инспектирование.

Благодаря проекту Binder [8], использующего инструмент jupyter-repo2docker, документ Jupyter Notebook из репозитория GitHub загружается в подготовленный контейнер Docker и становится доступным пользователям GitHub в интерактивном режиме. На создание контейнера уходит некоторое время, а после периода бездействия контейнер удаляется. Изменения, вносимые студентами в контейнерную версию, не затрагивают оригинальный репозиторий и могут рассматриваться как временные.

В качестве примера приведем один из репозиториев курса «Алгоритмы и структуры данных», разработанный для проведения практического занятия «Итерация и рекурсия» [9]. На главной странице репозитория имеются две ссылки для перехода на обычную и интерактивную версии учебного материала (рисунок 2).

Практическое занятие 01. Итерация и рекурсия

- Неинтерактивная версия: <https://github.com/NNTU-CS/ADS-class-01/blob/main/ADS-class-01.ipynb>
- Интерактивная версия: <https://mybinder.org/v2/gh/NNTU-CS/ADS-class-01/HEAD>

Рисунок 2 — Ссылки на две версии практикума

При переходе по ссылке интерактивной версии, студент попадает в среду, очень напоминающую интерфейс Google Colab, и, далее, получает возможность вносить изменения в серые ячейки с программным кодом (рисунок 3). Благодаря интерактивному режиму взаимодействия с материалом практикума, появляется уникальная возможность выполнять исследования по быстродействию различных алгоритмов, внося исправления в учебный код, что необходимо при рассмотрении вопросов асимптотической сложности алгоритмов. Еще одним вариантом может выступать задание «Найдите ошибку» в реализации алгоритма.

The screenshot shows a web-based Jupyter Notebook interface. The title bar indicates the file is 'ADS-class-01.ipynb'. The main content area displays the text 'Практическое занятие 01. Итерация и рекурсия' followed by an explanation of iteration and recursion. Below this, 'Задача 1. Вычисление факториала целого числа' is presented. The text explains that iteration is more memory-efficient than recursion. A formula for factorial is given: $n! = n * (n-1) * (n-2) * \dots * 1$. It then states that the following code uses an iterative solution with a loop. The code is in C++ and is shown in a grey code cell. The code includes `<iostream>`, defines a `fact1` function that calculates factorial iteratively, and calls `fact1(5)` with the output printed to the console. The output shows the memory address `@0x10974bc30` and the value `120`. Below the code, it mentions another iterative method for calculating factorial.

```
[7]: #include <iostream>

[2]: int fact1(int n) {
    int result=1;
    for(int i=n; i>=1; i--)
        result *=i;
    return result;
}
std::cout << fact1(5); // 5*4*3*2*1
120
[2]: @0x10974bc30

Есть другой способ итеративного вычисления факториала:

n! = 1 * 2 * ... * (n-1) * n
```

Рисунок 3 — Интерактивная версия практикума

В заключение можно отметить, что современные облачные технологии, технологии контейнерной виртуализации в сочетании с веб-технологиями позволяют разрабатывать интерактивные учебные курсы, в которых обучающиеся могут менять входные данные, вносить изменения в программный код,

добиваясь решения поставленных задач, не используя локально установленные среды разработки и библиотеки. Это способствует унификации решений поставленных задач и снимает с преподавателя необходимость консультации обучающихся по настройкам локальных инструментов.

Список литературы

1. *Штанюк, А. А.* Системы управления проектами для поддержки учебного процесса / А. А. Штанюк. Текст: непосредственный // Современные информационные технологии и ИТ-образование. 2012. № 8. С. 380–385.
2. *Винник, В. К.* Информационно-проектный метод при подготовке будущих специалистов в сфере информационных технологий (с использованием системы MOODLE) / В. К. Винник, А. А. Штанюк. Текст: электронный // Фундаментальные исследования. 2015. № 2, ч. 23. С. 5183–5186. URL: <https://fundamental-research.ru/ru/article/view?id=38178>.
3. *JupyterLab*: интерфейс для ноутбуков следующего поколения. Текст. Изображение: электронные // Jupyter. URL: <https://jupyter.org>.
4. *Агалаков, С. А.* Использование Jupyter notebook для проведения презентаций / С. А. Агалаков. Текст: непосредственный // Омские научные чтения–2020: материалы Четвертой Всероссийской научной конференции, Омск, 30 ноября – 05 декабря 2020 года. Омск: Омск. гос. ун-т им. Ф. М. Достоевского, 2020. С. 2181–2184.
5. *Тихонов, А. И.* Jupyter notebook – платформа для создания и применения в учебном процессе интерактивных расчетных приложений и электронных учебников / А. И. Тихонов. Текст: непосредственный // Современные образовательные Web-технологии в системе школьной и профессиональной подготовки: сборник статей участников Международной научно-практической конференции, Арзамас, 25–27 мая 2017 г. Арзамас: Арзамас. фил. Нац. исслед. Нижегород. гос. ун-та им. Н. И. Лобачевского, 2017. С. 126–132.
6. *Google Colab*. URL: <https://colab.research.google.com/drive/>. Текст. Изображение: электронные.
7. *GitHub* URL: <https://github.com>. Текст. Изображение: электронные.

8. *Binder* URL: <https://mybinder.org>. Текст. Изображение: электронные.
9. *Материал* к практикуму № 1 по курсу «Алгоритмы и структуры данных» // GitHub. URL: <https://github.com/NNTU-CS/ADS-class-01>. Текст. Изображение: электронные.