

нулевая гипотеза, согласно которой районный фактор несущественно влияет на объём потребляемой продукции.

Библиографический список

1. Аброськина, О. Обзор Российского рынка сыра / О. Аброськина // Российский продовольственный рынок. – 2016. - №2. – С.28-30.

2. Черняк М.Ю. Планирование и организация эксперимента: практикум для бакалавров направления подготовки 221700.62 «Стандартизация и метрология» / М.Ю. Черняк, М.С. Эльберг. – Сиб.гос.аэрокосмич. ун-т. – Красноярск, 2014. – 88 с.

УДК 311

ПОЧЕМУ УЧЕННЫЕ ДОЛЖНЫ НАУЧИТЬСЯ ПРОГРАММИРОВАТЬ НА PYTHON

Нагиева Роза, магистр кафедры статистики и эконометрики 124 группы Института экономики и управления АПК, ФГБОУ ВО РГАУ-МСХА имени К.А. Тимирязева, roza.nagieva@yandex.ru

***Аннотация:** в статье проводится анализ функций доступных для решения широкого круга задач в Python. Представлены особенности Python.*

***Ключевые слова:** Python, программное обеспечение, вычисления.*

Когда ученые поняли, что компьютеры могут облегчить их работу, они, возможно, были самыми восторженными первыми последователями этой технологии. По мере того как компьютеры становились все более доступными в 1950-х, 1960-х и 1970-х годах, ученые все больше осваивали их и разрабатывали программное обеспечение для выполнения длительных вычислений и автоматизации утомительных операций по сбору данных. Лишь немногие области были преобразованы так сильно, как Кристаллография, с помощью программ для фазирования прямых методов и уточнения наименьших квадратов, а также с помощью компьютеризированных инструментов. Пятьдесят лет спустя мысль о записи измерений на ленточный самописец кажется такой же причудливой, как поездка на работу в конном экипаже.

В настоящее время существует множество узкоспециализированных программ для порошковой дифракции, и практикующие специалисты также используют множество инструментов общего назначения, таких как электронные таблицы и пакеты обработки текстов. В то время как использование компьютеров и их мощность в науке выросли, есть ирония в том, что гораздо меньше ученых изучают навыки разработки программного обеспечения. Это большая потеря. В то время как существующие приложения могут выполнить совсем немного, в науке всегда есть простые задачи, которые никто не запрограммировал удобным образом. Кроме того, всегда есть новые идеи, которые следует попробовать. Также вызывает беспокойство вопрос о

том, кто будет писать следующее поколение научного программного обеспечения? Даже когда ученые объединяются с компьютерными специалистами для разработки программного обеспечения, это все равно очень полезно, когда у ученых есть хорошее понимание процесса программирования. Поэтому рекомендуется, чтобы больше ученых учились программировать. К счастью, не только компьютеры стали повсеместными, но и навыки, необходимые для изучения программирования, были упрощены, по крайней мере, с некоторыми компьютерными языками высокого уровня.

Существует множество языков программирования, причем каждый компьютерный язык был разработан таким образом, чтобы восполнить пробел в возможностях своих предшественников, но даже в этом случае каждый язык общего пользования имеет свои преимущества и недостатки, некоторые из которых будут рассмотрены ниже. Даже профессиональные программисты, как правило, специализируются на использовании небольшого числа языков, но поскольку у ученых значительно меньше времени для инвестиций, они обычно предпочитают овладеть одним языком, который может удовлетворить как можно больше потребностей.

Научные вычисления требуют, чтобы несколько типов задач выполнялись в сочетании: необходимо выполнить численный или символический анализ, обычно с научным программным пакетом, чтобы избежать изобретения колеса; результаты должны быть представлены пользователю, обычно с научной графикой; пользователь должен взаимодействовать с программой, как правило, в современных кодах через графический пользовательский интерфейс (GUI или веб-браузер); документация необходима для описания программного обеспечения и для пользователей, чтобы узнать, как использовать программу. Язык программирования Python способен выполнять все это и многое другое и таким образом удовлетворяет то, что нужно ученым (van Rossum, 1998). Кроме того, Python является одним из самых простых языков для изучения новичками, в то же время, являясь одной из лучших сред программирования для научных целей, таких как автоматизация, обработка изображений и численный анализ. Он также обеспечивает среду, в которой пользователи могут начать только с более простых возможностей, но, возможно, перейти к использованию более мощных функций, таких как объектно-ориентированное программирование, по мере роста навыков.

Python предлагает возможности для двух различных подходов к научным вычислениям: «числовые» и «символические» научные системы. Первый требует инструментов, специально предназначенных для обработки точных числовых данных (например, Matlab, Octave, R-language и другие высокоуровневые компьютерные языки), в то время как второй манипулирует неопределенными символьными выражениями или уравнениями (например, Mathematica и Maple). Поскольку все эти функции доступны в Python, он может быть применен для решения широкого круга задач. Новичок может легко освоить Python, чтобы очень быстро превратить свои идеи в программы, но если это необходимо для более крупного проекта, он может научиться создавать объектно-ориентированный код, оптимизировать код для скорости и

разработать сложную научную визуализацию со сложными графическими пользовательскими интерфейсами.

Кроме того, Python - это кросс-платформенный пакет программного обеспечения с открытым исходным кодом, который лицензируется под лицензией «Python Software Foundation License», что позволяет свободно распространять интерпретатор. Программы, написанные в нем, не несут никаких скрытых затрат на лицензирование. Это делает его идеальным для использования в классе, а также в лаборатории, так как тиражирование работы является краеугольным камнем научного метода. Ниже объясню многие особенности Python и объясню, почему он так ценен почти для всех аспектов научных вычислений, от быстрого и грязного преобразования данных формата до подгонки модели через автоматизацию приборов и даже теорию первых принципов. Сначала сравню Python с некоторыми другими языками программирования, обычно используемыми в научных вычислениях, а затем представим краткий обзор синтаксиса Python.

То, что делает Python столь ценным для научных вычислений, – это не только удобный для начинающих синтаксис Python, но и множество пакетов, которые позволяют выполнять многие общие задачи программирования в десятках строк кода, а не в сотнях или тысячах других языков. Опять же, есть тысячи таких пакетов, поэтому мы выделяем небольшой выбор из них, чтобы подчеркнуть, что делает Python таким ценным, и указать новичкам на некоторые из самых ценных ресурсов.

Python – это мощный язык программирования, хотя и достаточно простой, чтобы преподавать его на вводных курсах средней школы. Он может быть легко изучен, но все же предлагает огромную силу для профессиональной разработки программного обеспечения. Большое богатство научных пакетов, из которых только несколько были представлены здесь, показывает высокую ценность Python в руках ученых.

Библиографический список

1. Бэрри, Пол Изучаем программирование на Python / Пол Бэрри. - М.: Эксмо, 2016. - 332 с.

2. Васильев, А. Н. Python на примерах. Практический курс по программированию / А.Н. Васильев. - М.: Наука и техника, 2016. - 432 с.

3. Гуриков, С.Р. Основы алгоритмизации и программирования на Python / С.Р. Гуриков. - М.: Форум, 2018. - 991 с.

4. Гуриков, С.Р. Основы алгоритмизации и программирования на Python. Учебное пособие. Гриф МО РФ / С.Р. Гуриков. - М.: Инфра-М, Форум, 2018. - 707 с.

5. Златопольский, Д. М. Основы программирования на языке Python / Д.М. Златопольский. - М.: ДМК Пресс, 2017. - 277 с.