



ҚАЗАҚСТАН РЕСПУБЛИКАСЫ
ТҰҢҒЫШ ПРЕЗИДЕНТІ - ЕЛБАСЫНЫҢ ҚОРЫ

«ҒЫЛЫМ ЖӘНЕ БІЛІМ – 2017»

студенттер мен жас ғалымдардың
XII Халықаралық ғылыми конференциясының
БАЯНДАМАЛАР ЖИНАҒЫ

СБОРНИК МАТЕРИАЛОВ

XII Международной научной конференции
студентов и молодых ученых
«НАУКА И ОБРАЗОВАНИЕ – 2017»

PROCEEDINGS

of the XII International Scientific Conference
for students and young scholars
«SCIENCE AND EDUCATION - 2017»



14th April 2017, Astana



**ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ
Л.Н. ГУМИЛЕВ АТЫНДАҒЫ ЕУРАЗИЯ ҰЛТТЫҚ УНИВЕРСИТЕТІ**

**«Ғылым және білім - 2017»
студенттер мен жас ғалымдардың
XII Халықаралық ғылыми конференциясының
БАЯНДАМАЛАР ЖИНАҒЫ**

**СБОРНИК МАТЕРИАЛОВ
XII Международной научной конференции
студентов и молодых ученых
«Наука и образование - 2017»**

**PROCEEDINGS
of the XII International Scientific Conference
for students and young scholars
«Science and education - 2017»**

2017 жыл 14 сәуір

Астана

УДК 378

ББК 74.58

Ғ 96

Ғ 96

«Ғылым және білім – 2017» студенттер мен жас ғалымдардың XII Халықаралық ғылыми конференциясы = The XII International Scientific Conference for students and young scholars «Science and education - 2017» = XII Международная научная конференция студентов и молодых ученых «Наука и образование - 2017». – Астана: <http://www.eni.kz/ru/nauka/nauka-i-obrazovanie/>, 2017. – 7466 стр. (қазақша, орысша, ағылшынша).

ISBN 978-9965-31-827-6

Жинаққа студенттердің, магистранттардың, докторанттардың және жас ғалымдардың жаратылыстану-техникалық және гуманитарлық ғылымдардың өзекті мәселелері бойынша баяндамалары енгізілген.

The proceedings are the papers of students, undergraduates, doctoral students and young researchers on topical issues of natural and technical sciences and humanities.

В сборник вошли доклады студентов, магистрантов, докторантов и молодых ученых по актуальным вопросам естественно-технических и гуманитарных наук.

УДК 378

ББК 74.58

ISBN 978-9965-31-827-6

©Л.Н. Гумилев атындағы Еуразия
ұлттық университеті, 2017

корректировке в зависимости от социально-экономических и политических изменений.

При соблюдении всех указанных требований весь комплекс и его части обеспечат: а) отражение достижений науки, техники и технологии в других сферах, связанных с изучаемой дисциплиной (полиграфия, лингвистика, логика, литературоведение, экономика, журналистика, математика, химия, физика); б) последовательное изложение (без дублирования) учебного материала; в) использование современных методов, технологий и технических средств в образовательном процессе; г) рациональное распределение времени для изучения тем учебной дисциплины и организации самостоятельной работы студентов; д) профессиональную направленность образовательного процесса с учетом специфичных условий и востребованностью организаций - заказчиков кадров.

Список использованных источников

1. Лозицкий, В.Л. Электронный учебно-методический комплекс по дисциплинам социально- гуманитарного цикла. Научно-методические основы создания и системного применения / В.Л. Лозицкий. – Минск : РИВШ, 2012. – 224 с.
2. История книги и издательского дела : тематические тесты и задания для студентов специальности 1-47 01 01 «Издательское дело» / авт. сост. : В. И. Куликович, Д. П. Зылевич. – Минск : БГТУ, 2013. – 92 с.

УДК 004.432.42

ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ НА ПРИМЕРЕ PYTHON

Өмірзақ Ислам Аманжолұлы

islam.omirzak@gmail.com

Магистрант кафедры «Информатика», факультета информационных технологий,
ЕНУ им. Л.Н.Гумилева, Астана, Казахстан
Научный руководитель – Нурбеков А.Б., доктор PhD

Тенденция обучения программированию с раннего возраста становится все актуальнее с развитием информационных технологий. Осознание того, что обучение программированию не только облегчит, но и даст понять, как производятся вычисления в электронных устройствах, обеспечит устойчивое развитие обучающихся на протяжении всей жизни и способствует внедрению изучения программирования с начальной школы. В мировом образовательном сообществе намереваются сделать компьютерное программирование обязательным школьным предметом на всех уровнях системы образования. Мировые лидеры также являются сильными сторонниками расширения обучения программированию, и заявляют, что «необходимо чтобы все обучались программированию» [1].

Существует множество языков программирования, сред разработки и инструментов к ним, которые нелегко изучать без фундаментальных знаний. Для начинающих изучать программирование были созданы визуальные объектно-ориентированные среды программирования, такие как Scratch, BYOB и их аналоги, но, несмотря на это, нет понятливой комплексной методологии обучения программированию. В данной статье рассматривается возможность применения языка программирования Python в качестве основного языка для изучения, а также рекомендации по интеграции в процесс обучения.

С момента появления, портативные мобильные устройства стали приобретать все большую популярность среди детей дошкольного возраста, и в последние годы, быстро растет число детей, подвергающихся воздействию подобных устройств. Большинство детей учатся использовать эти электронные устройства до того, как они учатся читать и писать. В раннем возрасте, частично, они становятся компьютерно-грамотными, но не знают работы вычислительной техники и поэтому на данный момент многие родители и педагоги

поощряют интерес детей к технологиям, чтобы они могли понять, как такие устройства действительно работают и для чего нужен язык программирования. Педагоги все больше удостоверяются в том, что обучение программированию дадут детям навыки на протяжении всей жизни. Поскольку в большинстве развитых стран уровень безработицы среди молодежи является высоким, такие компьютерные навыки будут востребованы для того, чтобы ликвидировать разрыв между спросом и предложением на рынке труда, так как рынку необходимы специалисты, обладающие квалификацией и компьютерной грамотностью.

Проведя анализ учебных программ в мировом сообществе, можно констатировать тот факт, что информационно-коммуникационные технологии постепенно заменяются новым предметом, computing (вычисление), особенно на ранних этапах обучения. Это способствует преемственности и изменениям в обучении, открывает новые возможности, позволяет образовательным учреждениям возможность пересмотреть и усовершенствовать нынешние подходы в образовании, способствует разработке еще более увлекательной и продуманной программы, которая сможет ответить на вызовы информационного века. Введение данного предмета связано с тем, как компьютеры и компьютерные системы работают, и как они разработаны и запрограммированы. Учащиеся, изучающие вычисления, получают представление о вычислительных системах всех видов, независимо от того, есть ли компьютеры в данных системах или нет. Вычислительное мышление обеспечивает понимание во многих областях учебной программы и влияет на расширение кругозора, соприкасаясь с другими дисциплинами.

Почему так важно вычислительное мышление? Оно позволяет решать проблемы, разрабатывать системы и понимать силу и пределы человеческого и машинного интеллекта. Это навык, который дает возможность мыслить, который все ученики должны изучить и развивать в себе. Учащиеся, которые могут думать с вычислительной точки зрения, могут лучше осмысливать, понимать и использовать компьютерные технологии и поэтому лучше подготовлены к нынешним реалиям и в будущем.

Основное внимание в новой учебной программе, несомненно, уделяется программированию, а также другим аспектам информатики. С самого начала программирование было частью национальной учебной программы, но зачастую рассматривалось поверхностно. Информатика включает в себя методы и методы решения проблем и продвижения знаний, а также включает в себя четкий образ мышления и работы, который отличает её от других дисциплин. Любой базовый принцип можно преподавать или иллюстрировать, не полагаясь на использование конкретной технологии. Роль программирования в информатике аналогична роли практической работы в других науках - она обеспечивает мотивацию и контекст, в котором идеи воплощаются в жизнь.

Набор навыков программирования, включенных в обновленную программу выглядят следующим образом:

- Понимание и разработка алгоритмов, способность реализовать алгоритм с помощью языков программирования, использование данных, выполнений операций над данными, разработка и отладка созданных программ;
- Использование операторов условия, циклов при разработке программ, умение различать и использование различных типов переменных [2].

Соответственно, понадобятся инструменты, с помощью которых ученики могут начать программировать. Например, Scratch (MIT) предоставляет все инструменты, необходимые для удовлетворения требований к программированию новой учебной программы. Существуют альтернативы: Kodu - это богатая игровая среда, обеспечивающая графический «путь» к программированию; Logo имеет очень длинную историю как язык программирования высокого уровня, разработанный в 1967 году Сеймуром Пейпертом и Идит Харель в образовательных целях для обучения детей дошкольного и младшего школьного возраста основным концепциям программирования (рекурсии, расширяемости и пр.). Специалисты-практики вводят учеников в текстовое программирование с использованием Python.

Большинство специалистов в области программирования, считают, что Python - это лучший язык программирования для обучения новичков. Тем не менее Python был и является, одним из самых популярных языков программирования в мире. Каждый день, неосознанно, используется программное обеспечение, написанное с использованием Python. Python используется компаниями для написания разнообразных приложений. Google, NASA, Bank of America, Disney, CERN, YouTube, Mozilla, The Guardian - список компаний и организаций различных масштабов во всех секторах экономики, которые используют Python.

Почему язык программирования Python так популярен и актуален?

Рассмотрим ответ на этот вопрос с образовательной точки зрения. Разделим его на три пункта:

- Ресурсы. Имеется множество ресурсов для учащихся всех возрастов и уровней. Они варьируются от традиционных учебников до веб-сайтов, предлагающих курсы самостоятельного обучения в программировании на Python. Благодаря развитию проектов Raspberry Pi и One Laptop Per Child (OLPC) каждый может обеспечить доступное оборудование, работающее на Python.

- Дизайн языка. Python прост в освоении, довольно понятен и содержит множество библиотек, которые позволяют программистам создавать всевозможные приложения, адресованные различным доменам и видам деятельности. Удобен для чтения: если немного присмотреться, написанный код на языке Python будет понятен даже тем людям, которые не являются программистами (например, Python очень популярен среди ученых которые не занимаются программированием).

- Сообщество. Python имеет большое, разнообразное и активное сообщество, связанное с ним. Python Software Foundation (PSF) - это благотворительная организация под руководством сообщества, миссия которой заключается в продвижении, защите и продвижении языка программирования Python.

Рассмотрим второй пункт на примере простой программы. В исходном коде есть функция, называемая greet user, которая выводит приветствие:

```
def greet_user(username):  
    """Display a simple greeting."""  
    print("Hello, " + username.title() + "!")  
  
greet_user('jesse')
```

Вывод приветствия на экран:

```
Hello, Jesse!
```

Обратим внимание, что строки кода, реализующие функцию greet_user, имеют отступы. Кроме того, каждая инструкция в коде находится в отдельной строке. Код легко читается и понятен: очевидно, какие строки кода относятся друг к другу, согласно отступам.

В большинстве других компьютерных языков для обозначения области видимости используются синтаксические символы, а не отступы. Например, во многих языках, таких как Java, JavaScript и C, для этой цели используются фигурные скобки и точки с запятой. Также используется встроенная функция title, которая выводит параметр с заглавной буквы и встроенная функция print для вывода текста на экран. Для этого не нужно писать библиотеки в исходном коде, в отличие от Java, C++ [3].

Достаточно базовых основных понятий и знаний, необходимых для написания кода на Python. Кроме того, эти понятия легки в освоении, использовании и запоминании, что является основополагающей характеристикой Python.

Рассмотрим следующие два примера.

В объектно-ориентированных языках программирования, для создания списка вещей, используются конструкции с разными именами, такие как массивы, векторы и т.д. В Python для этих целей используется список. Например, список дел написанных на Python:

```
todo_list = ['Shopping', 'Fix broken gutter', 'Mow the lawn']
```

Этот код присваивает списку заданные в квадратных скобках значения (в данном примере это строки символов, содержащие слова, которые описывают задачи в списке дел) объекту с именем `todo_list` (который можно позже использовать для ссылки на список, а также на конкретный элемент из списка, посредством индекса).

Для создания базы данных, которая позволит хранить и искать именованные значения (базовое хранилище ключей / значений), в других языках, как правило, должны использоваться конструкции, называемые структурами, ассоциативными массивами, картами или таблицами. В Python используется нечто, называемое словарем. Рассмотрим словарь данных небольшого количества столиц нескольких стран:

```
capital_cities = {  
    'China': 'Beijing',  
    'Germany': 'Berlin',  
    'Greece': 'Athens',  
    'Russia': 'Moscow',  
    'United Kingdom': 'London',  
}
```

В данном примере объекту `capital_cities` присвоили словарь. Для обращения к столице определенной страны, указывается название страны в квадратных скобках рядом с объектом с именем `capital_cities`:

```
capital_cities['China']  
'Beijing'
```

Многие языки программирования не имеют структур данных, которые работают аналогично спискам и словарям в Python. Преимуществом Python является то, что: у него есть полезные структуры данных как основная часть языка, он дает им очевидные имена и делает их чрезвычайно удобными в использовании. Такая полезность, простота и ясность являются еще одним примером устранения барьеров на пути к программированию.

Создатель Python, Ван Россум, осознал потенциальную роль языка в мире образования. В 1999 году он опубликовал свою позицию по этому вопросу через предложение о проекте под названием «Компьютерное программирование для всех: скаутская экспедиция для программистов завтра». В первых параграфах кратко описываются его перспективы:

«В семидесятых годах научно-исследовательский центр Хегох ставил перед собой следующий вопрос: «Сможем ли мы иметь компьютер на каждом столе?» Доступность персональных компьютеров показало, что это возможно, но эти компьютеры не находятся в полной власти пользователей. Современные компьютеры не используют всю свою мощь: среднестатистический пользователь компьютера, как правило, может изменять только ограниченный набор параметров, настраиваемых через «настройки компьютера» (может называться по-разному), и только опытные программисты в состоянии настраивать всё остальное.»

Из этого следует следующий вопрос: «Что произойдет, если пользователи смогут полностью настраивать (т.е. программировать) свой собственный компьютер?» Исследователи с нетерпением ждут будущего где каждый пользователь компьютера сможет «открыть потенциал» своего компьютера и внести улучшения в приложения внутри. Они полагают, что, в конечном итоге это коренным образом изменит характер программных средств и средств разработки программного обеспечения.

Разработчики языка Python планировали решить это через три компонента:

1. Разработать новый учебный план, подходящий для школьников и студентов.
2. Создавать лучшие, более простые в использовании инструменты для разработки и анализа программ.
3. Создать сообщество пользователей вокруг всего вышеперечисленного, поощряя обратную связь и самопомощь.

К сожалению, проект так и не был завершен из-за отсутствия финансирования. Но

вопрос остается открытым «а что, если?». Как мог развиваться Python в контексте образования, если бы проект был завершен на всех трех запланированных компонентах?

В любом случае, это еще одно доказательство того, что Python всегда учитывал образование как основное направление. Это предложения были дальновидными, учитывая недавние изменения в отношении к учебным программам и содействию программированию.

Тем не менее, такие образовательные начинания не остались незамеченными.

Проекты, связанные с вычислительной техникой и обучением, успешно использовали Python для больших результатов. Например, проект «Один ноутбук на ребенка» (One Laptop Per Child) имеет следующую цель:

Стремление предоставить каждому ребенку прочный, недорогой и маломощный подключенный ноутбук. С этой целью мы разработали оборудование, контент и программное обеспечение для совместного, радостного и самостоятельного обучения. Имея доступ к этому типу инструментов, дети занимаются своим образованием, учатся, совместно используют и создают вместе. Они становятся связанными друг с другом, с миром и с более светлым будущим.

Пользовательский интерфейс и приложения для OLPC были написаны на Python. Более 2,5 миллиона детей и преподавателей в 42 странах имеют такие ноутбуки.

Пожалуй, самым известным и успешным проектом в области образования в истории является Raspberry Pi (в настоящее время на нем представлено более 5 миллионов устройств). Все это является доказательством того, что Python является лучшим языком для изучения программирования [4].

В системе основного среднего образования Республике Казахстан информатика преподается с 5-го класса, введение в программирование начинается в 6ом классе, где учащимся объясняют алгоритмы, их виды, и т.д. В 7-11 классах учащиеся проходят языки программирования (преимущественно Паскаль), проводятся региональные поэтапные олимпиады по информатике. Языки C++, Java и т.д. преимущественно преподаются в системе высшего образования, хотя в специализированных школах (напр. КТЛ, НИШ) в качестве факультативов. Имеется позитивный опыт по внедрению изучения предмета робототехники в общеобразовательных школах. Но в целом, методика преподавания остается от преподавания зарубежных стран из-за недостаточного финансирования, низкой заинтересованностью педагогов (в плане повышения квалификации, изучению новых языков программирования). Нет комплексной системы обучения программированию, но схеме «школа-вуз-производство», а также последовательности в обучении программированию.

Из своего опыта, во время прохождения практики в казахско-турецком лицее г. Астаны для одаренных мальчиков в 2016 году, могу констатировать, что в данном образовательном учреждении, с помощью образовательного фонда KATEV, учителя информатики проходят стажировки, курсы повышения квалификации в образовательном центре USTAZ данного фонда, куда приглашаются преподаватели зарубежных стран для обмена опытом и улучшения образовательного процесса. Например, в прошлом году Джоном Маккарти (John McCarthy, профессор Madonna University) проводились семинары по внедрению Project-Based Learning (PBL). Также, в 7-11 классах существует связь в обучении программированию на основе визуально объектно-ориентированной среды программирования Scratch, простых систем автоматики и робототехники Arduino, и языка программирования C++. Также внутри фонда ведутся обсуждения по поводу внедрения языка Python вместо C++, так как Python и Scratch дополняют друг друга в образовательном процессе, а также возможно заменить Arduino на Raspberry Pi в целях унификации обучения программированию.

Исходя из этого, считаю, что при положительном эффекте в системе среднего образования, вполне разумно использовать язык программирования Python для ознакомления с программированием, а также разработать технологию практико-ориентированного обучения на примере Python.

Список использованных источников

1. Heath N. Programming set to be core of new computing classes for English kids// ZDNet.com, 2013, P.1-2.
2. Berry M. Computing in the national curriculum. A guide for primary teachers// British Library, 2013, P. 5-14.
3. Matthes E. Python Crash Course. – San Francisco, No Starch Press, 2016, P 167-168.
4. Tollervey N.H. Python in education. – CA, O'Reilly Media, 2015, P 14-20

УДК 371.004

РАЗРАБОТКА ЦИФРОВОГО АНИМАЦИОННОГО КОНТЕНТА ПО ИКТ ДЛЯ ВУЗОВ НА АНГЛИЙСКОМ ЯЗЫКЕ

Рамазанова Диана Атантаевна

Diana_260694@mail.ru

Евразийский Национальный Университет имени Л.Н. Гумилева,
Астана, Казахстан

Научный руководитель – Абильдинова Г.М

Сегодня автоматизация любой сферы становится одним из наиболее востребованных и перспективных направлений, как в научно-производственной сфере, так и в сфере образования.

Одной из важнейших задач, стоящих перед казахстанской системой образования, является обеспечение доступности и качества образовательного процесса, итогом которого должно стать формирование конкурентоспособного выпускника.

Подобные технологии активно применяются для передачи информации и обеспечения взаимодействия преподавателя и студента в современных системах открытого и дистанционного образования.

«Технология» относится к термину греческого происхождения, что в переводе означает науку, а также совокупность методов и приемов обработки или переработки сырья, материалов, изделий и преобразования их в предметы потребления. Современное понимание этого слова включает и применение научных и инженерных знаний для решения практических задач. В таком случае информационными и телекоммуникационными технологиями можно считать такие технологии, которые направлены на обработку и преобразование информации.

В настоящее время учебный процесс в высшей школе продолжает сохранять объяснительно-иллюстративный характер. Тогда как одно из ведущих положений теории деятельности для эффективного обучения предполагает такую организацию деятельности, при которой студент сам оперирует учебным содержанием, и только в этом случае оно усваивается осознанно и прочно, а также идет процесс развития интеллекта обучающегося. Парадигма состоит в том, что студент должен учиться сам, а преподаватель – осуществлять мотивационное управление его обучением, т.е. мотивировать, организовывать, координировать, консультировать и контролировать. Поэтому перед казахстанским образованием встала проблема поиска технологии обучения, позволяющей решить эту задачу. Такой технологией как раз и является модульное обучение, ибо оно базируется на позициях деятельного, активного, гибкого подхода к построению педагогического процесса.

Модульное обучение - способ организации учебного процесса на основе блочно-модульного представления учебной информации. Каждый модуль имеет свою интегрирующую дидактическую цель. В модули входят крупные блоки учебного содержания.

Современные учебно-методические комплексы дисциплин в ВУЗах составлены