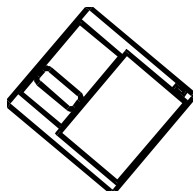


Учредитель –
Учреждение «Главный
информационно-
аналитический центр
Министерства образования
Республики Беларусь»

Свидетельство о регистрации
№662
выдано 28.10.2010 г.
Министерством информации
Республики Беларусь



220088, г. Минск,
ул. Захарова, 59

<http://giac.unibel.by>

**Ежеквартальный
научно-методический журнал**

(приказом ВАК РБ от 31 марта 2016 г.
№ 84 включен в перечень научных изданий
Республики Беларусь для опубликования
результатов диссертационных исследований)

1(77) 2016

Информатизация образования

Издается с IV квартала 1995 года

Редакционный совет

Н.И. Листопад (главный редактор)
А.Н. Курбацкий (предс. ред. совета)
С.И. Ладутько (зам. гл. редактора)
Е.Н. Кишкурно (отв. секретарь)
М.М. Ковалев
В.Н. Ярмолик
М.К. Буза
Б.Н. Паньшин
В.В. Басько
М.Г. Зеков
В.И. Дравица
С.В. Енин
Д.И. Пунько

СОДЕРЖАНИЕ

К уроку информатики

О.А. Засим

Развитие критического мышления у учащихся на учебных занятиях по информатике **3**

Из опыта работы

М.С. Долинский

Элементы теории чисел: решето Эратосфена **11**

Е.С. Рогальский

Использование облачных сервисов для создания электронного конспекта студента **21**

Программно-технические средства

В.С. Осерко, З.В. Пунчик

Методические аспекты обучения работе с многопользовательской базой данных **31**

С.В. Корженевич, П.И. Личевский

Использование программы Plickers в образовательном процессе **39**

А.А. Шелестович

Kodu Game Lab как способ развития базовых основ программирования в игровой форме у школьников младшего и среднего возраста **42**

Научные публикации

Н.И. Листопад, И.А. Карук, А.А. Хайдер

Алгоритмы поиска кратчайшего пути и их модификация **48**

Е.М. Шевчик-Гирис

Полисемия термина «информационно-коммуникационная культура» **64**

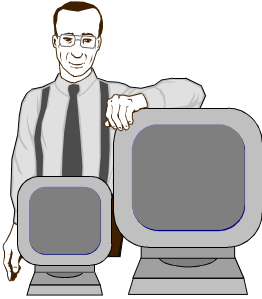
Н.М. Люцко

Становление информационно-аналитического обеспечения научной деятельности учреждений высшего образования в условиях университетских библиотек **74**

Н.А. Балашенко

Информационные технологии в генетике **84**

К УРОКУ ИНФОРМАТИКИ



***О.А. Засим**, директор гимназии, учитель информатики высшей квалификационной категории государственного учреждения образования «Гимназия №1 имени Ф.Я. Перца г. Пинска»*

Развитие критического мышления у учащихся на учебных занятиях по информатике

Теле- и видео сети, Интернет сделали возможным обмен информацией по всему земному шару за несколько минут. Так, по словам М. Маклюэна, возникает «глобальная деревня, над которой раскрыт электронный зонт» [1, С.5]. «Размеры земного шара мысленно уменьшаются до размеров небольшой деревни, где жители знают все друг о друге, где возможен доступ практически к любой информации» [5, С.5].

Как защитить подрастающее поколение от неправдивой и негативной информации, которая распространяется в Интернете со скоростью звука? Раньше, когда дети приходили из школы, родители у них спрашивали: «Как прошел день в школе?». А сейчас надо интересоваться: «Как прошел день в Интернете?». Границы между виртуальным и реальным стираются. По словам известного шведского медиапедагога Ерана Андерсена, происходит то, что в социальных медиа напоминает взрыв. У учителей и родителей учащихся не всегда хватает времени и возможностей, чтобы это все отследить. Есть только один выход: развивать критическое мышление у самих

учащихся, учить их разбираться, где в виртуальном мире правда, а где ложь. Школьное образование просто обязано помочь решить возникшую проблему. Это можно делать как во внеурочное время, так и на учебных занятиях. Остановлюсь более подробно, как можно это осуществить на учебных занятиях по информатике. По моему мнению, это можно реализовать следующим образом:

- проводить уроки по технологии критического мышления либо с элементами названной выше технологии;
- организовывать учебные занятия по информатике в форме «перевернутых уроков».

Остановлюсь на первом направлении. «Под критическим мышлением понимают процесс оценки достоверности, точности или ценности чего-либо, способность искать и находить причины и альтернативные точки зрения, воспринимать ситуацию в целом и изменять свою позицию на основе фактов и аргументов» [2]. Технология развития критического мышления дает ученику:

- повышение эффективности восприятия информации;
- повышение интереса как к изучаемому материалу, так и к самому процессу обучения;
- умение критически мыслить;
- умение работать в сотрудничестве с другими;
- повышение качества знаний учащихся.

На уроках информатики можно использовать следующие методические приемы развития критического мышления:

«Методический прием записной книжки Хефеле» заключается в том, что за неделю до коллективного обсуждения учащимся дается задание и сообщается тема, по которой они должны в течение недели делать записи.

Вот пример использования приема «Методика записной книжки Хефеле».

Тема: «Компьютерные презентации», VII класс.

При подготовке к проектной работе учащимся за неделю дается задание: в течение недели собрать материал про известного человека нашего города (например, человека, имя которого носит наша гимназия) по плану:

1. Детство и юность.
2. Чем известен.
3. Его заслуги.
4. Какой вклад внес в развитие города и гимназии.

Дети собирают материал: проводят интервью, анкетирование, посещают места, где работал и жил интересующий их человек. На уроке работают с найденной информацией (с текстовой, графической и др.).

Методический прием синектики основан на превращении непривычного в привычное, а привычного в непривычное.

Например, при изучении темы в VI классе «Обработка графической информации» учащиеся получают задание: как в наше время мог бы выглядеть Кощей Бессмертный.

Разбивка на кластеры. «Кластер (от англ. – cluster – гроздь) – это способ графической организации материала, позволяющий сделать наглядными те мыслительные процессы, которые происходят при погружении в тот или иной текст» [3]. Кластер является отражением нелинейной формы мышления. Иногда такой способ называют «наглядным мозговым штурмом». Последовательность действий при построении кластера проста и логична:

1. Посередине чистого листа (классной доски) необходимо написать ключевое слово или тезис, который является «сердцем» текста.

2. Вокруг «накидать» слова или предложения, выражающие идеи, факты, образы, подходящие для данной темы.

3. По мере записи, появившиеся слова соединяются прямыми линиями с ключевым понятием. У каждого из «спутников», в свою очередь, тоже появляются «спутники», устанавливаются новые логические связи.

В итоге получается структура, которая графически отображает размышления, определяет информационное поле данного текста.

Учащимся IX класса можно предложить составить кластер по теме: «Web-ресурсы сети Интернет» (см. рис. 1).

Синквейн-способ. Происходит от французского слова «sing» – пять. Это стихотворение, состоящее из пяти строк. «Синквейн – это не простое стихотворение, а стихотворение, написанное по следующим правилам:

- 1 строка – одно существительное, выражающее главную тему синквейна.

- 2 строка – два прилагательных, выражающих главную мысль.

- 3 строка – три глагола, описывающие действия в рамках темы.

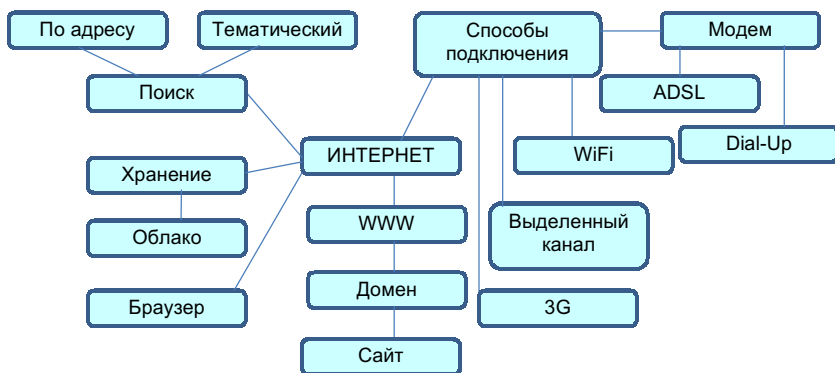


Рис. 1. Web-ресурсы сети Интернет

4 строка – фраза, несущая определенный смысл.

5 строка – заключение в форме существительного (ассоциация с первым словом)» [4].

Используется как способ синтеза материала. Лаконичность формы развивает способность резюмировать информацию, излагать мысль в нескольких значимых словах, емких и кратких выражениях. Сиквейн может быть предложен, как индивидуальное самостоятельное задание; для работы в парах; реже как коллективное творчество. Границы предметной области зависят от гибкости воображения учителя. Обычно синквейн используется на стадии рефлексии, хотя может быть дан и как нетрадиционная форма в начале учебного занятия.

Как показывает опыт, синквейны могут быть полезны в качестве:

- 1) инструмента для синтезирования сложной информации;
- 2) способа оценки понятийного багажа учащихся;
- 3) средства развития творческой выразительности.

Вот несколько примеров синквейнов, которые учащиеся составляли на уроках информатики.

Диск

Хрупкий, гладкий
Принимает, хранит, отдает
Мой помощник в хранении информации
Память

Интернет

Интересный, стремительный
Хранит, ищет, находит

Сеть для поиска информации Паутина

Развитию критического мышления также способствуют нестандартные формы проведения учебных занятий. Можно провести учебное занятие по информатике в виде *урока – суда* над компьютером. Выбираются действующие лица: Судья, Прокурор, Адвокат. Подсудимые: Процессор, Оперативная Память, Видеокарта, Материнская плата, Винчестер. Прокурор обвиняет устройства компьютера в ненужности, а Адвокат старается доказать обратное. Каждое устройство подробно рассказывает, зачем оно необходимо. Присутствующие могут задавать вопросы с разрешения судьи.

Применяя технологию критического мышления на учебных занятиях по информатике, можно сделать обучение более эффективным в плане пробуждения интереса к предмету, развития критического осмысления учениками получаемой в процессе обучения и жизненного опыта информации, осознанной работы с изучаемым материалом, умения обобщать, проводить рефлексию своей деятельности, подводить итоги.

Одним из современных способов развития критического мышления является форма смешанного обучения – «перевернутый класс». Перевернутым становится сам процесс обучения. «Перевернутый класс» – это модель обучения, когда теоретический материал изучается посредством информационных и коммуникационных технологий самостоятельно» [5].

Современные технические средства и средства коммуникации предоставляют огромные возможности для общения вне класса учителя с учениками, учеников между собой, самостоятельного поиска информации. Обучение возможно построить так, что изучив теоретический материал дома, ученик приходит в класс, чтобы выяснить непонятные вопросы, которые остались после домашней подготовки, и закрепить пройденное.

Вначале записанное видео по материалам нового урока раздается ученикам (выкладывается на общий ресурс, например в облако). В качестве домашнего задания учащиеся смотрят короткие видео-лекции, самостоятельно проходят новый теоретический материал. Ученик может смотреть в удобное время и несколько раз, чтобы разобраться в изучаемом материале. Мотивированный ученик может прослушать объяснение столько раз, сколько ему нужно, чтобы понять материал. Попутно он может обратиться к учебнику и дополнитель-

ным ресурсам, тем самым сравнить, сопоставить фактический материал. Учитель к этому ролику прилагает несколько вопросов или небольшой тест для мониторинга усвоения учебного материала. По ответам детей он видит уровень понимания нового материала.

От того, насколько ответственно ученик будет выполнять домашнее задание, будет зависеть его успешность на уроке. Подобная форма организации урока позволит привлечь учеников к реальной деятельности на уроке, а не к скучному записыванию лекций за учителем. Для этого нужно поменять содержание домашней работы и работы на уроке. Вместо выполнения обычных домашних упражнений ученику предоставляется доступ к электронным ресурсам. Получив начальные знания и сформировав свое представление о предмете изучения, учащиеся, придя в класс, переходят к активной форме обучения, т.е. к анализу конкретной ситуации, решению проблемных задач, выполнению практического задания.

Однако можно пойти по более сложному пути: учащимся не дается готовое видео, они сами ищут материал по заявленной теме в Интернете. В этом случае это создает большие предпосылки для развития критического мышления. Во-первых, учащиеся учатся искать достоверную информацию, проверять ее на подлинность, используя несколько источников. Во-вторых, отличать научные знания от антинаучных. Конечно, это требует больших сил, времени и затрат энергии, но пользы от такой подготовки домашних заданий будет гораздо больше.

По учебному предмету «Информатика» есть много тем, которые лучше было бы провести при активной деятельности учащихся на уроке и не тратить урочное время на объяснение материала, т.е. провести учебное занятие в форме перевернутого урока. В VII классе изучается тема: «Устройство компьютера». Как показывает практика, очень мало детей когда-либо видели и держали в руках процессор, оперативную память, другие детали системного блока. Очень часто уроки по этой теме проходят без активной деятельности учащихся. Учителя могут показать плакат, стоящий на столе компьютер и прочесть текст в учебнике. Учащиеся, как правило, воспринимают компьютер как некую железную коробку. Процессор, оперативная память и прочие детали компьютера остаются для них абстрактными понятиями. Идея собрать компьютер вызывает живой интерес у подростков.

Я ушел от традиционного урока по этой теме и предложил учащимся ознакомиться с устройством компьютера дома по самостоятельно найденным видеоматериалам, а на уроке закрепить свои знания с помощью практической работы по сборке системного блока. Проведение такой практической работы позволило учащимся внимательно изучить все детали системного блока, их функции, определить их местонахождение-

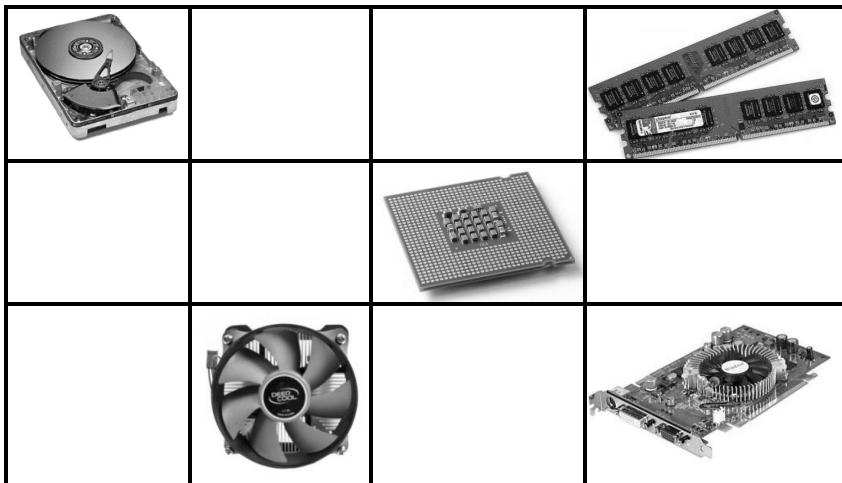


Рис. 2

ние. При обобщении материала по данной теме, я использовал игру «Лото» (см. рис. 2). Учащиеся вытягивают бочонки с написанными на них числами. Некоторым числам соответствуют детали системного блока. Если учащиеся вытягивают именно такие числа, значит им необходимо рассказать об определенной детали системного блока.

Проводя такие уроки, я отметил следующие их преимущества:

- У учащихся повышается интерес и мотивация к изучаемому предмету.
- Ученики работают в своем темпе. При просмотре видео можно поставить на паузу, перемотать и пересмотреть видео лекцию. Это позволяет ученикам распределить время учебы по своему усмотрению. Можно пропустить то, что кажется понятным и повторить то, что оказалась сложным.

- Материалы урока доступны всем учащимся и тем, кто пропустил уроки по каким-либо причинам.
- Возможность обучения для учащихся вне аудитории в удобное время.
- Учащиеся помогают друг другу в учебе. Некоторые ученики понимают материал быстрее и в «перевернутом классе» учитель может назначить их наставниками, помощниками.
- Учитель выступает в роли координатора познавательной деятельности учащихся.

Выстраивая работу на учебных занятиях подобным образом, мы сможем научить подрастающее поколение «хорошо ориентироваться в медиапространстве, т.е. оно будет уметь работать с различными источниками информации, уметь эту информацию искать, систематизировать, критически ее воспринимать, уметь пользоваться медиатехникой» [6, 7].

Литература

1. Маклюэн, М. Галактика Гуттенберга: становление человека печатающего/ Перевод И.О. Тюриной. – М.: Академический Проект: Фонд «Мир», 2005. – С.5.

2. Критическое мышление. / [Электронный ресурс]. – Режим доступа: <http://wiki.pskovedu.ru/index.php> – Дата доступа: 31.05.2016.

3. [Электронный ресурс]. – Режим доступа: <http://www.russkiymir.ru/sites/default/russkiymir/ru/biblioteka-russkogomira/zadaniya/Bohanov-Imperator-Aleksandr-III.pdf> – Дата доступа: 01.06.2016.

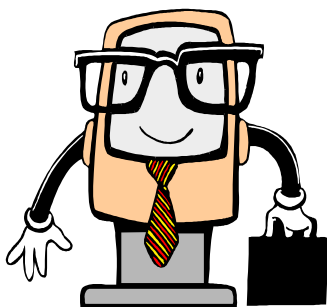
4. Составление синквейна с примерами. / [Электронный ресурс]. – Режим доступа: <http://lewe.ru/sostavlenie-sinkvejnas-primerami/> – Дата доступа: 30.05.2016.

5. [Электронный ресурс]. – Режим доступа: <http://metodbank.sri-kiz.ru/DswMedia/urokotehnologiiperevernutiyklass.pdf> – Дата доступа: 01.06.2016.

6. Засим, О.А. Медиакультура: сущность, значение, внедрение в систему современного образования / О.А. Засим // Адукацыя і выхаванне. – 2015. – № 8. – С.7.

Статья поступила 10.04.2016





ИЗ ОПЫТА РАБОТЫ

М.С. Долинский, к.т.н., доцент кафедры математических проблем управления Гомельского государственного университета им. Ф. Скорины

Элементы теории чисел: решето Эратосфена

Введение

Автор много лет занимается обучением программированию школьников разных возрастов и первокурсников математического факультета (специальности: «Программное обеспечение информационных технологий» и «Прикладная математика») [1-21]. Все это время автор занимался созданием материалов для самостоятельного изучения школьниками и студентами, стараясь представить материал в как можно более простой, наглядной и понятной форме. В данной статье приводится пример такого материала для обучения решению задач по информатике на тему «Решето Эратосфена». Такой материал может быть интересен для преподавателей как в качестве иллюстрации методики обучения, так и по содержанию. В то же время, автору представляется, что этот материал может оказаться весьма полезным и интересным и для школьников, и для студентов, занимающихся самообучением. Всем заинтересованным предлагается следующий порядок работы: откладывать статью в сторону и пытаться самостоятельно выполнить предлагаемое задание, первый раз –

после прочтения условия задачи, второй раз – после прочтения указаний к решению.

Решето Эратосфена

Так называют следующий способ получения ряда простых чисел.

Из ряда чисел

2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14 ... вычеркивают кратные двум;

4, 6, 8, 10, 12, ... – кратные трем:

6, 9, 12, 15, ... – кратные пяти:

10, 15, 20, 25, 30, ... – кратные семи:

14, 21, 28, 35, 42, 49, ... и т. д.

Таким образом, все составные числа будут просеяны, и останутся только простые числа 2, 3, 5, 7, 11, 13 ... (Энциклопедический словарь Ф.А. Брокгауза и И.А. Ефрона. – С.-Пб.: ЕСБЕ, 1890-1907).

Ниже представлена процедура Resheto, которая вышеописанным способом находит все простые числа в диапазоне от 2 до MaxX.

```
const
  MaxX = 4000000
Procedure Resheto;
var
  i,j : longint;
begin
  for i:=2 to MaxX do x[i]:=1;
  for i:=2 to trunc(sqrt(MaxX)) do
    if x[i]=1
      then begin
        j:=i;
        while (j+i)<=MaxX do
          begin
            j:=j+i;
            x[j]:=0;
          end;
        end;
end;
```

Пояснения. Мы хотим создать массив x из MaxX элементов. Если число i ($2 \leq i \leq \text{MaxX}$) простое, то $x[i]=1$, а иначе (число i не простое (то есть составное)), $x[i]=0$.

Вначале мы прописываем все элементы числом 1. Затем, в цикле, начиная со второго элемента. Если он не равен 1, вычеркиваем (заносим число 0 на соответствующие позиции j) все элементы, кратные i , то есть $i+i$, $i+i+i$, $i+i+i+i$, ... (то есть $2*i$, $3*i$, $4*i$...) пока этот индекс ($k*i$) не вышел за границу массива $MaxX$.

Математиками доказано, для того, чтобы получить все простые числа в диапазоне от 2 до N , достаточно в этом алгоритме вычеркивать числа кратные всем простым числам от 2 до корня квадратного из N (\sqrt{N}). Поскольку граница цикла `for` в языке программирования Паскаль должна быть целой, используется функция `trunc` – для того, чтобы превратить корень из N в целое число отбрасыванием его дробной части.

После выполнения этой процедуры в массиве x хранятся пометки всех простых чисел от 2 до N . Для того, чтобы узнать, является ли число Z ($2 \leq Z \leq MaxX$) простым, достаточно проверить значение массива x в позиции Z . Если $x[Z]=1$, значит число Z – простое.

Задача 07_JanB. Qualified Primes [Kolstad/Ho, 2007]

Фермер Джон начал пометку коров последовательными простыми числами. Бесси заметила это и задумалась о вхождении различных цифр в эти метки. Помогите Бесси определить количество простых чисел в интервале (включая концы) $A..B$ ($1 \leq A \leq B \leq 4\,000\,000$; $B \leq A + 1\,000\,000$; один тест имеет $B \leq A + 2\,000\,000$), которые содержат некоторую данную цифру D . Простым называется положительное целое число, которое имеет ровно 2 делителя (1 и себя). Первые простые числа: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29.

Формат ввода:

* Строка 1: Три разделенных пробелом целых числа: A , B , D

Пример ввода (файл `qprime.in`):

10 15 3

Пояснения: сколько простых чисел в диапазоне 10..15 содержит цифру 3?

Формат вывода:

* Строка 1: Количество простых чисел в указанном диапазоне, которые содержат цифру D .

Пример вывода (файл `qprime.out`):

1

Пояснения: только число 13 в данном диапазоне содержит цифру '3'.

Для решения задачи вначале с помощью вышеописанной процедуры Resheto построим массив X, содержащий пометки, является ли число I простым ($x[i]=1$) или составным ($x[i]=0$). А затем переберем все числа в интервале от A до B и проверим каждое, является ли оно простым и содержит ли цифру D. Для последней проверки используется функция Yes(I,D), которая проверяет, содержится ли цифра D в числе I переводом числа в строку и вызовом встроенной функции pos.

```
{07_JanB. Qualified Primes}
const
  MaxX = 4000000;
var
  x : array [2..MaxX] of longint;
  i,num,a,b : longint;
  d : char;
Procedure Resheto;
var
  i,j : longint;
begin
  for i:=2 to MaxX do x[i]:=1;
  for i:=2 to trunc(sqrt(MaxX)) do
    if x[i]=1
      then begin
        j:=i;
        while (j+i)<=MaxX do
          begin
            j:=j+i;
            x[j]:=0;
          end;
        end;
end;

function Yes(i:longint; d:char):boolean;
var
  s : string;
begin
  str(i,s);
  if pos(d,s)<>0
    then Yes:=true
    else Yes:=false;
end;
```

```

begin
  assign(input,'qprime.in'); reset(input);
  assign(output,'qprime.out'); rewrite(output);
  readln(a,b,d,d); Resheto;
  num:=0;
  for i:=a to b do
    if (x[i]=1) and Yes(i,d) then inc(num);
  writeln(num);
  close(input); close(output);
end.

```

Задача 05_OctB. Max Factor [Hal Burch, 2005]

Чтобы улучшить организацию своей фермы, фермер Джон пометил каждую из своих N ($1 \leq N \leq 5\,000$) коров уникальным номером в диапазоне от 1 до 20 000. К несчастью, он не предполагал, что коровам некоторые номера нравятся больше других. В частности, корова, чей номер имеет больший простой делитель, имеет больший социальный статус. Напомним, что простое число – это такое число, которое не имеет других делителей, кроме себя и 1. Например, число 7 – простое, а число 6 – не простое, поскольку имеет делители 2 и 3.

Вам дано N ($1 \leq N \leq 5\,000$) чисел в диапазоне от 1 до 20 000.

Определите такое из них, которое имеет наибольший простой делитель.

Формат ввода:

* Строка 1: Одно целое число N

* Строки 2.. $N+1$: номера, которые нужно проанализировать, по одному в строке

Пример ввода (файл bigfact.in):

```

4
36
38
40
42

```

Формат вывода:

* Строка 1: Целое число с самым большим простым делителем. Если таких чисел несколько, выведите то из них, которое раньше появилось в исходном списке чисел.

Пример вывода (файл bigfact.out):

```

38

```

Пояснение к выводу: 19 – самый большой делитель числа 38. Никакое из других чисел не имеет простого делителя больше, чем 19.

И снова для решения задачи вначале построим массив пометок X, указывающих, является ли число l простым ($x[i]=1$) или составным ($x[i]=0$). Затем считываем числа, заданные по условию задачи, и для каждого из них с помощью процедуры MaxFactor ищем наибольший простой делитель соответствующего числа. По ходу запоминаем максимальный из найденных наибольших простых делителей и число ему соответствующее, которое и выводим в качестве ответа по завершении ввода всех заданных чисел.

```
{05_OctB. Max Factor}
const
  MaxX = 20000;
var
  x : array [2..MaxX] of longint;
  i,n,max,a,num,mf : longint;

Procedure Resheto;
var
  i,j : longint;
begin
  for i:=2 to MaxX do x[i]:=1;
  for i:=2 to trunc(sqrt(MaxX)) do
    if x[i]=1
      then begin
        j:=i;
        while (j+i)<=MaxX do
          begin
            j:=j+i;
            x[j]:=0;
          end;
        end;
end;

Procedure MaxFactor(a:longint; var mf:longint);
var
  j : longint;
begin
  j:=a div 2;
  while (j>1) and (((a mod j)<>0) or (x[j]=0)) do dec(j);
  if j>1
```



```

        then mf:=j
        else mf:=a;
end;

begin
    assign(input,'bigfact.in'); reset(input);
    assign(output,'bigfact.out'); rewrite(output);
    readln(N); max:=1; num:=1; Resheto;
    for i:=1 to N do
        begin
            readln(a); MaxFactor(a,mf);
            if mf>max then begin num:=a; max:=mf; end;
        end;
    writeln(num);
    close(input); close(output);
end.

```

Пояснение к процедуре MaxFactor(a,f), которая для заданного числа a находит его наибольший простой делитель mf.

```

Procedure MaxFactor(a:longint; var mf:longint);
var
    j : longint;
begin
    j:=a div 2;
    while (j>1) and (((a mod j)<>0) or (x[j]=0)) do dec(j);
    if j>1
        then mf:=j
        else mf:=a;
end;

```

В цикле от половины числа a (делителем не может быть число, которое больше половины a) проверяем все числа j до 2 включительно. Если a не делится на j или j не простое, берем число на 1 меньше.

Если встретилось такое число j, что a делится на j и j – простое (x[j]=1), то это j и есть наибольший простой делитель. Иначе – наибольший простой делитель есть само число a (других простых делителей не найдено).

Заключение

В данной статье приведен материал для обучения решению задач по информатике на тему «Решето Эратосфена».

Технической основой методики является разработанная инструментальная система дистанционного обучения (Distance Learning Belarus – <http://dl.gsu.by>). Все задачи, приведенные в статье, могут быть сданы в курсе «Методы алгоритмизации».

Литература

1. Долинский, М.С. Об опыте подготовки школьников Гомельской области к республиканским и международным олимпиадам по информатике / М.С. Долинский // Информатизация образования. – 2009. – № 1(54). – С.29-40.

2. Долинский, М.С. Система интернет-курсов дифференцированного обучения программированию школьников и студентов / М.С. Долинский, М.А. Кугейко // Информатизация образования. – 2010. – № 1(58). – С.58-68.

3. Долинский, М.С. Как учить думать школьников и студентов? / М.С. Долинский, М.А. Кугейко // Информатизация образования. – 2010. – № 2(59). – С.62-72.

4. Долинский, М.С. Технология развивающего дифференцированного обучения программированию младших школьников «с чистого листа» / М.С. Долинский, М.А. Кугейко // Информатизация образования. – 2010. – № 3(60). – С.12-20.

5. Долинский, М.С. Интернет-курс «Базовое программирование» как средство подготовки к областным олимпиадам по информатике / М.С. Долинский, М.А. Кугейко // Информатизация образования. – 2010. – № 4(61). – С.3-15.

6. Долинский, М.С. Развитие мышления младших школьников на основе флеш-заданий на рисование, раскраску и конструирование в системе DL.GSU.BY / М.С. Долинский, Ю.В. Решетько, М.А. Кугейко // Информатизация образования. – 2011. – № 1(62). – С.24-35.

7. Долинский, М.С. Какими должны быть задачи на олимпиадах по информатике / М.С. Долинский, М.А. Кугейко // Информатизация образования. – 2011. – № 1(62). – С.68-76.

8. Долинский, М.С. Флеш-шаблоны для создания заданий развивающего обучения / М.С. Долинский, Ю.В. Решетько, М.А. Долинская // Информатизация образования. – 2011. – № 2(63). – С.14-28.

9. Долинский, М.С. Конструирование интерактивных флеш-заданий на развитие мышления / М.С. Долинский, Ю.В. Решетько, М.А. Долинская // Информатизация образования. – 2011. – № 3(64). – С.21-33.

10. Долинский, М.С. Конструирование интерактивных флеш-заданий на развитие мышления на базе произвольных картинок / М.С. Долинский, Ю.В. Решетько, М.А. Долинская // Информатизация образования. – 2011. – № 4(65). – С.3-14.

11. Долинский, М.С. Конструирование интерактивных флеш-заданий на базе собственных танов / М.С. Долинский, Ю.В. Решетько, Н.С. Лебедько // Информатизация образования. – 2012. – № 1(66). – С.24-34.

12. Долинский, М.С. Конструктор интерактивных флеш-заданий как открытая система для создания электронных учебных пособий / М.С. Долинский, Ю.В. Решетько, М.А. Долинская, Н.С. Лебедько // Информатизация образования. – 2012. – № 2(67). – С.35-45.

13. Долинский, М.С. Электронное учебное пособие «Математика. Начальная школа» / М.С. Долинский, Ю.В. Решетько, Н.С. Лебедько // Информатизация образования. – 2012. – № 3(68). – С.30-42.

14. Долинский, М.С. Создание электронных учебных пособий для вузовских дисциплин с помощью конструктора флеш-заданий / М.С. Долинский, Ю.В. Решетько // Информатизация образования. – 2012. – № 4(69). – С.34-45.

15. Долинский, М.С. Интерактивная анимация в электронных учебных пособиях, создаваемых с помощью конструктора флеш-заданий / М.С. Долинский, Ю.В. Решетько, М.А. Долинская // Информатизация образования. – 2013. – № 1(70). – С.30-38.

16. Долинский, М.С. Учебный интернет-курс и перманентный интернет-конкурс «Математика 1-8 кл.» / М.С. Долинский, Ю.В. Решетько, М.А. Долинская // Информатизация образования. – 2013. – № 2(71). – С.38-47.

17. Долинский, М.С. Концептуальные основы и практика сквозного развивающего обучения информатике и программированию от детского сада до вуза / М.С. Долинский, Ю.В. Решетько, М.А. Долинская // Информатизация образования. – 2013. – № 3(72). – С.16-25.

18. Долинский, М.С. Об одном подходе к обучению программированию на первом курсе / М.С. Долинский, М.А. Долинская // Информатизация образования. – 2014. – № 1(73). – С.32-41

19. Долинский, М.С. Использование форума при обучении программированию первокурсников / М.С. Долинский // Информатизация образования. – 2014. – № 2(74). – С.22-34.

20. Долинский, М.С. Элементы теории чисел: системы счисления / М.С. Долинский // Информатизация образования. – 2015. – № 1(75). – С.14-28.

21. Долинский, М.С. Элементы теории чисел: битовая обработка / М.С. Долинский // Информатизация образования. – 2015. – № 2(76). – С.3-15.

Статья поступила 04.01.2016



*Е.С. Рогальский, старший преподаватель
кафедры компьютерных технологий и си-
стем Белорусского государственного уни-
верситета*

Использование облачных сервисов для создания электронного конспекта студента

Введение

В настоящее время все больший интерес среди студентов приобретают цифровые, в том числе и облачные технологии. Облачные технологии – это модель предоставления повсеместного и удобного сетевого доступа к общему пулу конфигурируемых вычислительных ресурсов (например, к серверам, приложениям и другим ресурсам сети, таким как системы хранения и сервисы), которые могут быть быстро предоставлены и освобождены с минимальными усилиями по управлению и необходимости взаимодействия с провайдером [1].

Этот интерес обусловлен возможностью использования электронных учебных курсов и материалов электронных библиотек, доступностью в качестве приложений для смартфонов [2], наличием различных сервисов, в частности, программирования как сервиса, платформы (для создания приложений) как сервиса, хранения как сервиса на образовательных облачных ресурсах и других [3]. Важным фактором является то, что свои персональные данные, в частности конспекты, отчеты по лабораторным работам, курсовые проекты и другие учебные материалы в объеме от 2 до 5 (иногда даже 10 и более) Гб студенты имеют возможность расположить на облачных ресурсах без оплаты. Это популярные облачные ресурсы, такие как Яндекс Диск, DropBox, Облако @Mail.ru и некоторые другие.

Такое положение вещей приводит к тому, что многие студенты пытаются переосмыслить организацию своей учебной деятельности в университетах, особенно там, где имеются приватные (корпоративные, частные) и гибридные облака, электронные (цифровые) библиотеки, активно развивается ди-

станционное (цифровое) обучение, предоставлен доступ к электронным учебным курсам, широко используются на практике различные образовательные технологии (вебинары, удаленная работа, используются технологии Web2.0, облачные вычисления и др.). Некоторые студенты идут дальше – они создают свои «персональные» электронные конспекты на базе предлагаемых в электронных библиотеках электронных учебных курсов.

Процесс цифровой ориентации современной молодежи обусловлен факторами, рассмотренными на Всемирном саммите по информационному обществу [4]. Поколение «цифровых аборигенов» видит многие вещи иначе, в том числе учебные пособия и литературные источники, предлагаемые им для обучения [5]. В такой ситуации логично дать студентам относительную свободу по форме сохранения информации, предоставив преподавателю прерогативу содержательного наполнения предлагаемых учебных курсов. Решение этой проблемы предлагается в данной статье как технология использования облачных сервисов для создания персонально-ориентированного электронного конспекта студента.

1. Электронный конспект

Электронный конспект – это своеобразная «бридж технология», являющаяся, по сути дела, мостом между преподавателем и восприятием учебного материала студентом. Электронный конспект совмещает технические возможности компьютерной и видеотехники в представлении учебного материала с живым общением лектора с аудиторией, хотя возможно создание электронного конспекта и без лектора в оффлайн режиме. Техническими средствами для этих целей обладает практически каждый студент. Это ноутбук, планшет или смартфон с большим экраном (таблоид).

Для создания информационной избыточности в лекции должны сочетаться и использоваться возможности различных уровней мышления человека [6]: предметно-чувственного, понятийно-логического и образно-эмоционального. Наглядная фотографическая, графическая, знаковая информация предоставляет нам вещи, объекты, предметы изучения. Удельный вклад такого мышления у студентов довольно высок. Наглядность в лекции обязательна, она помогает запомнить материал, облегчает понимание более сложных вещей. Образно-эмо-

циональное описание может способствовать пониманию наиболее трудного объяснительного комментария, тем самым способствуя развитию в мышлении студента еще одной наглядности – наглядности математических выводов или стройных логических умозаключений.

Все перечисленное входит в состав методического инструментария преподавателя и способствует повышению уровня усвоения материала студентами, позволяя подать материал таким способом и в такой форме, чтобы он «сам запомнился». В результате возникает потребность интеграции в контент конспекта технологии эффективного использования (обучения), или, как ее называют, когнитивной составляющей, то есть такой технологии обучения, которая делает этот процесс удобным, доступным и понятным, активизирующим необходимые психомоторные механизмы памяти.

2. Этапы построения электронного конспекта

При создании электронного конспекта можно выделить следующие этапы:

1. Этап аудиторный;
2. Этап домашней обработки лекции;
3. Этап подготовки к экзамену;

и, если студент уделяет повышенное внимание к данному материалу, *этап самостоятельной разработки темы* с использованием дополнительных материалов по изучаемому предмету –

4. Этап самостоятельной разработки темы.

Рассмотрим 1-й, аудиторный этап. На этом этапе студент работает над созданием электронного конспекта, понимая, какие требования необходимо учесть, чтобы дальнейшая работа над конспектом допускала максимальную (формальную) автоматизацию (использование программы, ориентированной на технологию электронного обучения – e-learning). Для этого рассмотрим план электронного конспекта, или в более доступной формулировке – требования с точки зрения программной реализации.

Для этого будем каждую текущую страницу электронного конспекта разбивать на две части, расположенные, соответственно слева и справа. Информацию, в каждую из частей, будем вносить по строго определенным правилам. Перейдем к рассмотрению этих правил:

В «Левой» части электронного конспекта расположена «Формальная часть конспекта». С помощью символов, выбранных студентом из банка эвристических символов программы, обозначаются категории персонального эвристического кодирования: математические выражения, аббревиатуры (принятые сокращения), ключевые слова, ярлык (иконка, рисунок) раздела, графы и другие объекты – активаторы памяти (подсказки). Набор используемых символов каждый студент формирует самостоятельно, выбирая из банка эвристических символов программы, в зависимости от индивидуальной потребности его персонального кодирования. Этот набор при приобретении студентом навыков такого конспектирования существенно оптимизируется и стабилизируется (индивидуально) как набор минимально необходимых объектов.

В «Правой» части электронного конспекта расположена «Вербальная часть конспекта». Это словесное описание всех элементов, в том числе включающее и текст лекции, и определения, а иногда неполный (отрывочный) пояснительный материал, требующий дополнительной работы студента.

Итог аудиторного этапа работы с электронным конспектом – прообраз файла, который требует восстановления, или, если попытаться сформулировать более точно, редактирования на этапе домашней обработки лекции. На этом этапе реализуется обучающая технология, позволяющая тренировать память и контролировать усвоение материала.

В качестве примера рассмотрим макет программной реализации электронного конспекта. Изначально вся информация будет разбиваться на некоторые блоки (модули, части или определения), что формально можно назвать планом конспекта или структурированием исходного файла. Студент сам определяет, какие элементы из предлагаемого программой набора эвристического кодирования использовать в качестве активатора памяти при редактировании (преобразовании) данного раздела конспекта (прообраза). Проведя таким образом требуемую редакцию (восстановление) прообраза, студент получает файл лекции, который он может создать и разместить как приложение [7] на облачном ресурсе, или как локальную версию на персональном компьютере. Далее с полученным файлом студент имеет возможность продолжить работу на этапах 2, 3 или 4 (если у него такая потребность возникает). Также он может сравнить «свою»






версию с итогами работы других студентов по данной теме. Студенты могут также провести «электронные публичные чтения», отобрав в итоговый конспект лучшую, на взгляд учебной группы, версию электронного конспекта. Такую работу можно аппроксимировать как еще один, 5-й этап работы с электронным конспектом: *этап коллективного редактирования электронного конспекта*.

Этот вид деятельности могут использовать и преподаватели при проведении практических занятий по теме. Нельзя не отметить и еще один результат – преподаватель имеет возможность таким образом влиять на качество своих будущих лекций, так как он имеет возможность оценить *практический результат восприятия лекций студентами* и работать над их редактированием в нужном для студентов направлении. Но не следует забывать, что главное назначение электронного конспекта – это повышение качества знаний студента, сокращение затрат времени на подготовку к экзаменам и информирование об уровне подготовки. Поэтому рассмотрим деятельность студента на этапе подготовки к экзамену. Нам поможет в этом рисунок 1, на котором представлен фрагмент работы с электронным конспектом¹.

При работе с электронным конспектом вниманию студента предлагается «Левая» часть (формальная), а «Правая» часть на монитор при этом не выводится. Если студент, работая с «Левой» частью электронного конспекта (формальной), знает то или иное определение, он помечает его значком «палец вверх»². Тогда, в «Правой» части, ему не выводится информация об этом определении. Если студент испытывает затруднения при ответе на вопрос, или сомневается в правильности ответа, он нажимает значок «палец вниз», и в вербальной части появляется определение с полным описанием данного вопроса. По окончании работы с электронным конспектом программа информирует, как студент освоил данную информацию. Если этот результат меньше 50% (или другого, заданного студентом уровня тестирования), то программа предлагает повторно ознакомиться

¹ На рис. 1 представлена одна из реализуемых форм диалога со студентом, соответствующая режиму контроля знаний (этапы 2,3 и 4).

² Это может быть любой символ из банка эвристических символов программы (по выбору студента).

Формальная	Вербальная
ЭК 	<p>Электронный конспект студента. Состоит из формальной и вербальной части.</p>
АСУ 	<p>Автоматизированная система управления – комплекс аппаратных и программных средств, а также персонала, предназначенный для управления различными процессами в рамках технологического процесса, производства, предприятия</p>
ЦОД 	<p>Центр обработки данных или «ЦОД» представляет собой централизованную вычислительную систему с достаточно высоким уровнем производительности и высокой готовностью для решения задач предприятия.</p>
ADSL 	<p>Asymmetric Digital Subscriber Line (асимметричная цифровая абонентская линия). Асимметрия заключается в том, что передача данных от абонента осуществляется медленнее, чем к нему.</p>
СУБД (DBMS) 	<p>Система управления базой данных (data base management system).</p>

Результат

Отвѣтов: 3 из 5

Процент правильных ответов: 60%

Назад
Следующий

Рис. 1. Деятельность студента на этапе подготовки к экзамену

с материалом. Когда студент достигает заданного уровня, он может продолжить работу с электронным конспектом с целью повышения уровня владения материалом путем задания более высокого порога тестирования или со следующими разделами. Студент, используя электронный конспект, сокращает затраты времени на подготовку к экзаменам, повышает качество своих знаний и имеет объективную информацию об уровне своей подготовки. Такая организация самостоятельной деятельности студентов позволяет существенно повысить уровень подготовки за счет использования современных облачных образовательных технологий.

3. Защита авторских прав и монетизация

Переходя к данному, заключительному разделу статьи, хочется отметить еще одно, немаловажное свойство предлагаемой обучающей технологии. У нас в периодической, учебно-методической, научной и специальной литературе [8, 9] уделяется достаточно внимания критике подходов, анализу перспектив и отставанию, а также анализу причин этого отставания в области электронного обучения, отсутствию необходимых кадров и финансирования. Но крайне мало работ, где приводятся реальные методы разработки продуктов электронного обучения – автоматизированных обучающих систем, программ – виртуальных агентов, тьюторов с искусственным интеллектом и т.д. Причины создавшейся ситуации известны, и вряд ли нас ждут какие-либо изменения в ближайшем будущем. Значит, мало разработать и предложить перспективный продукт, необходимо чтобы он содержал в своей структуре элементы, позволяющие зарабатывать на своей реализации, то есть элементы монетизации. Для продвижения проекта на рынок образовательных услуг его необходимо оформить, как стартап с целью привлечения потенциальных инвесторов. Для этого следует решить следующие практические задачи:

- Обеспечить необходимую рекламу возможностей и положительных качеств предлагаемого программного продукта. Это означает, что рекламный ролик, размещенный на профильных сайтах и/или в СМИ, должен содержать информацию об эффективности педагогических технологий, которые интегрированы в проект, чтобы потенциальный пользователь пони-

мал, каким образом достигается результат от использования этого программного продукта.

- Обеспечить соблюдение авторских прав, исключающих несанкционированное копирование и распространение с целью продажи пиратских копий. Это обеспечивается режимом доступа через авторизацию с персональным кодом к облачному ресурсу – приложению, позволяющему работать с электронным конспектом – приложением на облачном ресурсе.

- Прозрачность при организации переводов денег в электронном формате на счет официальных дилеров продукта. Оплата может производиться с привлечением кредитных (банковских) электронных карт и терминальных переводов в системе ЕРИП, почтовых и банковских переводов по системам «Стриж», «Рядом», «ЮНИСТРИМ», «ЗОЛОТАЯ КОРОНА», «Contact», дорожным чекам и т.д. Результат – получение персонального (оригинального) кода для авторизации доступа.

4. Заключение

Подводя итог, выделим основные результаты:

- При создании электронного конспекта студент обладает достаточной свободой для самореализации в плане «авторства» формы персонального средства обучения, реализует основной принцип «Умных технологий» – «я так хочу».

- Положительный эффект достигается интеграцией в контент конспекта технологии эффективного использования, то есть такой технологии обучения, которая делает этот процесс удобным, доступным и когнитивным, активирующим необходимые психомоторные механизмы памяти. Эффект достигается в сочетании с сокращением времени на подготовку к экзаменам, повышением качества знаний студента и получением объективной информации об уровне своей подготовки.

- Студент сам разрабатывает собственную форму электронного конспекта, осмысленно использует активаторы памяти (подсказки) – набор (алфавит) используемых символов из банка эвристических символов программы. Это также элемент творчества и индивидуальности электронного конспекта, так как каждый студент использует такой набор самостоятельно, в зависимости от индивидуальной потребности его эвристического кодирования.

- Преподаватель имеет возможность влиять на качество своих будущих лекций, так как он имеет возможность оценить практический результат восприятия лекций студентами и работать над редактированием их в нужном для студентов направлении.

- Уделено внимание продвижению проекта на рынке образовательных услуг, защите авторских прав, и монетизации проекта. Все это делает предложенный метод разработки электронного конспекта интересным для студентов, преподавателей и инвесторов. Метод реализован в виде программы, которую планируется разместить на облачном ресурсе.

Литература

1. Макаров, С.В. За «облачные вычисления»! Облачные вычисления как модель эффективного предоставления современных компьютерных услуг. // Креативная экономика. – 2010. – № 8. – С.114-121.

2. Левин А. Android на планшетах и смартфонах. – СПб.: Питер, 2013. – 224 с.

3. Рогальский, Е.С. Облачные технологии и их роль в развитии электронного обучения // Исследования наукограда. – 2014. – № 1. – С.42-49.

4. Всемирный Саммит по информационному обществу, 10-12 дек. 2003 г., Женева / Рос. ком. Прогр. ЮНЕСКО «Информация для всех», Рос. библ. ас-соц., Рос. нац. б-ка; сост. Е.И. Кузьмин, В.Р. Фирсов. – СПб.: Рос. нац. б-ка, 2004. – 135 с.

5. Логинова, А.В. Электронное обучение: причины незаинтересованности поколения «цифровых аборигенов» / А.В. Логинова // Молодой ученый. – 2015. – № 10. – С.1212-1215.

6. Пospelов, Д.А. Фантазия или наука: на пути к искусственному интеллекту. – М.: Наука, 1982. – 226 с.

7. Felker/Android Application Development For Dummies // Wiley Publishing, – Москва, 2010. – 364 с.

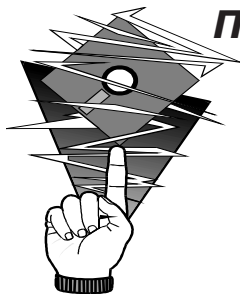
8. Фарино, К.С. Использование электронных средств обучения в образовательном процессе. // Настаўніцкая газета. – 17 жніўня 2010 г. – № 101. – С.5-8.

9. Смолин О.Н. Аналитическая записка и предложения по внесению изменений в нормативные правовые акты, под-

готовленные рабочей группой Экспертного Совета по вопросам электронного обучения и информационным технологиям в образовании Комитета Государственной Думы по образованию. // 09.10.2008. – № 5.2-21/2660. – Председателю Правительства Российской Федерации В.В. Путину.

Статья поступила 22.03.2016





ПРОГРАММНО-ТЕХНИЧЕСКИЕ СРЕДСТВА

В.С. Оскерко, к.э.н., доцент кафедры информационных технологий Белорусского государственного экономического университета,

З.В. Пунчик, к.социолог.н., доцент кафедры экономической информатики Белорусского государственного экономического университета

Методические аспекты обучения работе с многопользовательской базой данных

Обосновывается необходимость освоения специалистами технологии работы с многопользовательской БД в архитектуре «клиент/сервер». Кратко описывается эта архитектура и указываются ее преимущества. Излагаются методы реализации вышеуказанной технологии с помощью клиентского приложения Access. Отмечаются трудности создания в среде Access SQL-запросов к БД на сервере. Во избежание их для работы с этой БД предлагается использовать клиентское приложение Query Analyzer и указываются его достоинства, выражающиеся в разнообразии функциональных возможностей. Подход к обучению технологии работы с многопользовательской БД с помощью указанных клиентских приложений освещается в двух аспектах: как извлечь информацию из БД для анализа и как внести в БД изменения. Говорится о необходимости дать в учебном процессе представление о работе с многопользовательской БД с помощью Интернет-браузера и возможности его реализации через создание web-приложения в Access 2013.

Высокий динамизм процессов в обществе, науке, технике, производстве влечет экспоненциальный рост объема информации, что усложняет ее поиск, приводит к дублированию научных исследований, экспериментов, технологий, затрудняет принятие эффективных решений. Для преодоления этих проблем разрабатываются автоматизированные информационные системы (АИС).

Будущему специалисту придется работать в среде некоторой АИС, ядром которой является многопользовательская база данных (БД). Поэтому обязательным элементом профессиональной подготовки специалистов должно быть обучение работе с многопользовательской БД.

В настоящее время широко распространены компьютерные сети с архитектурой «клиент/сервер». В этой архитектуре на сервере сети размещается многопользовательская БД и устанавливается сервер БД – программный компонент, обеспечивающий хранение больших объемов информации, ее обработку и представление пользователям в сетевом режиме, а на компьютерах-клиентах – клиентские приложения. Функции клиентских приложений и сервера баз данных при доступе к данным базы определенным образом разделены.

Клиентское приложение по запросу пользователя на извлечение информации из БД или на внесение изменений в нее формирует SQL-запрос к БД на сервере. Сервер БД обеспечивает интерпретацию запроса, его выполнение и, при необходимости, формирование результата запроса и его пересылку по сети на клиентский компьютер. Клиентское приложение интерпретирует результат запроса необходимым образом и представляет его пользователю. Сервер БД также управляет целостностью базы данных, обеспечивает одновременную безопасную и отказоустойчивую многопользовательскую работу с одними и теми же данными, выполняет хранение и резервное копирование данных, реализует бизнес-правила с помощью ограничений, утверждений, хранимых процедур и триггеров.

Технология работы с многопользовательской БД в архитектуре «клиент/сервер» имеет ряд достоинств: снижение трафика; уменьшение потребности клиентских приложений в оперативной памяти; высвобождение на клиентских компьютерах значительного объема дискового пространства; существенное повышение безопасности БД.

Указанные достоинства архитектуры «клиент/сервер» являются стимулирующими факторами ее массового распространения, что обуславливает необходимость овладения будущими специалистами теоретическими знаниями о механизмах доступа к внешним базам на сервере и практическими навыками формирования запросов к многопользовательской БД в указанной архитектуре.

Технология формирования запросов во многом определяется используемым программным обеспечением. В локальных сетях вузов, как правило, в качестве сервера баз данных используется Microsoft SQL Server, а клиентского приложения – система управления базами данных (СУБД) Microsoft Access. В этом случае обучающимся можно продемонстрировать технологию доступа к данным на сервере сети в СУБД Access. В среде Access она может быть реализована двояко:

- путем установки связи с таблицами БД на сервере;
- путем формирования запросов к БД на сервере на диалекте языка SQL в Access – Microsoft Jet SQL.

Оба метода требуют создания уникальной для каждого пользователя ссылки на внешнюю БД – пользовательского источника данных. Его создание осуществляется с помощью Мастера ODBC (открытый интерфейс доступа к данным, разработанный компанией Microsoft), который запускается с Панели управления, открываемой из Главного меню Windows. После запуска Мастера необходимо выбрать драйвер, для которого задается источник, (например, SQL Server) и задать: имя пользовательского источника данных; имя сервера, на котором размещена многопользовательская БД; имя пользователя и пароль для входа в сервер; проверку подлинности учетной записи; имя БД.

Если применяется первый метод, то запросы могут создаваться с помощью визуальных средств конструирования запросов СУБД.

Для реализации второго метода необходимо знать функциональные возможности и синтаксис мощной SQL-команды SELECT и уметь создавать, редактировать и выполнять запросы к серверу в режиме SQL. Для формирования навыков создания SQL-запросов на извлечение из БД информации необходимо рассмотреть примеры таких запросов с различными критериями отбора, содержащими выражения с

операторами сравнения, логическими операторами, со специальными операторами (IN, BETWEEN, IS NULL, LIKE), а также примеры на вычисления в запросах (в том числе с использованием статистических функций), на создание вложенных запросов и др. При выполнении запросов к серверу указывается созданный ранее пользовательский источник данных, имя пользователя и пароль для входа в сервер.

Если при формировании SQL-запроса допущены ошибки, то их поиск и исправление возлагается на пользователя, так как информации о локализации ошибок и рекомендаций по их устранению СУБД Access практически не выводит.

В этой связи обучающихся целесообразно ознакомить с технологией доступа к данным на сервере с помощью специализированного клиентского приложения, в котором указанный недостаток преодолен. В качестве такого приложения можно использовать компонент Microsoft SQL Server – Query Analyzer. Это клиентское приложение позволяет формировать многофункциональные запросы на диалекте Transact-SQL. Так, в среде Query Analyzer можно создавать запросы выбора и запросы на внесение изменений в БД на сервере и выполнять их. При этом можно: проверять синтаксис команд после их набора; выполнять выделенную часть сформированного запроса; использовать шаблоны команд и редактировать их согласно собственным требованиям; применять Браузер объектов БД для упрощения процесса построения запросов; отображать результат запроса рядом с командой Transact-SQL; работать с запросами в нескольких окнах запроса; выводить на печать содержимое и области ввода команды, и области результата запроса; представлять результат запроса в различных форматах с целью возможности его обработки другими приложениями и др.

Пример SQL-запроса с результатом его выполнения в окне Query Analyzer приведен на рис. 1.

При работе с многопользовательской БД приходится не только извлекать информацию из нее для анализа, но и вносить в нее изменения. Освоение данной возможности частично может быть реализовано через задачу импорта-экспорта объектов БД в технологии «клиент/сервер» с помощью клиентских приложений Access и Import and Export Data (компонента Microsoft SQL Server). Для этой цели авторами осуществлена постановка задачи по имитации действий:

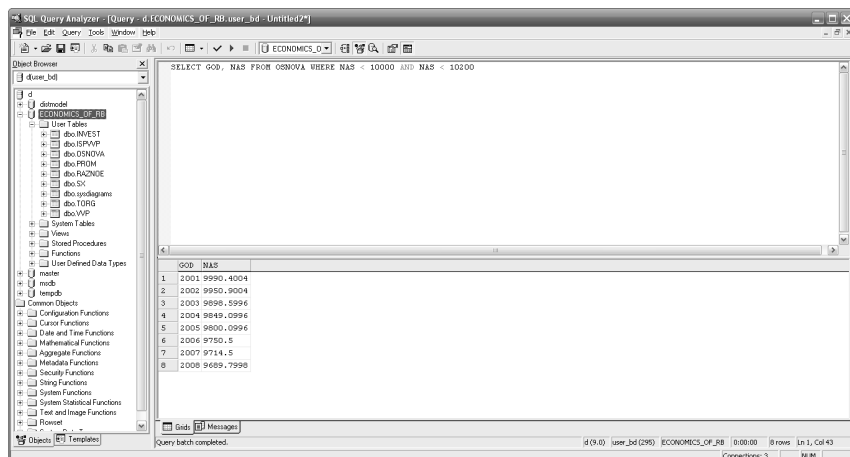


Рис. 1. SQL-запрос с результатом его выполнения в окне Query Analyzer

- секретаря деканата (клиента) по расчету среднего балла студентов по итогам экзаменационной сессии на основании справочника стипендий, созданного в БД Spravochnik на сервере сети, и записи расчетных данных в БД Stipendia;
- бухгалтера (клиента) по начислению стипендий на основании расчетных данных секретаря деканата, которую он извлекает из БД Stipendia с сервера.

Обучающиеся, в роли секретаря деканата (читающие данные БД и записывающие в БД) наделяются правом доступа к БД Spravochnik «только для чтения» и правом доступа к БД Stipendia – «можно изменять», а в роли бухгалтера – только правом доступа к БД Stipendia «можно изменять».

Выбранная в данной задаче предметная область понятна обучающимся и при наличии соответствующего методического обеспечения [2] технология импорта-экспорта объектов БД в архитектуре «клиент/сервер» легко усваивается.

Для обучения технологии формирования запросов к многопользовательской БД необходимо решить вопрос о создании и размещении на сервере сети учебной БД и присвоении обучающимся права доступа к ней «только для чтения». Учебную БД целесообразно наполнить реальной информацией из предметной области, вызывающей профессиональный интерес у будущих специалистов, и это будет сти-

мулировать приобретение практических навыков формирования запросов к многопользовательской БД в архитектуре «клиент/сервер».

Реализация авторами методики обучения формированию запросов к БД на сервере сети с помощью клиентских приложений Access и Query Analyzer на примере созданной ими на основе статистической отчетности БД «Социально-экономическое развитие Республики Беларусь» показала, что обучающиеся быстро осваивают инструментальные средства этих приложений при наличии соответствующего методического обеспечения [3] и реально видят практическое применение данной технологии. Ежегодное обновление вышеуказанной БД позволяет студентам использовать ее информацию о численности населения, о валовом внутреннем продукте и его использовании, о развитии отраслей народного хозяйства и др. для экономического анализа при выполнении научных, курсовых и дипломных работ.

Для демонстрации современной тенденции в развитии технологий БД, проявляющейся в их интеграции с WEB, целесообразно дать обучающимся представление о работе с многопользовательской БД через Интернет. Версия СУБД Access 2013 позволяет создавать web-приложение – БД, опубликованную на web-страницах сайта Microsoft SharePoint. С этой БД пользователям можно работать через Интернет коллективно с помощью web-браузера, при этом не устанавливая СУБД Access на компьютерах пользователей.

Для создания и публикации web-приложения понадобится Access 2013 и сервер SharePoint Server 2013 с поддержкой служб Access, установленный в сети учебного заведения. При создании веб-приложения в SharePoint Server, размещаемого в локальной сети, Access создает БД на SQL Server 2012, который определяется администратором SharePoint [1, С.426].

Создание web-приложения в среде Access 2013 требует указания его имени и выбора его расположения в Интернете (рис. 2).

Добавление таблиц в БД web-приложения возможно путем использования их шаблонов, создания «с нуля» при помощи Конструктора таблиц, импорта таблиц из БД Access, из файлов MS Excel и любых других источников, которые используют для доступа к данным интерфейс ODBC. Одновре-

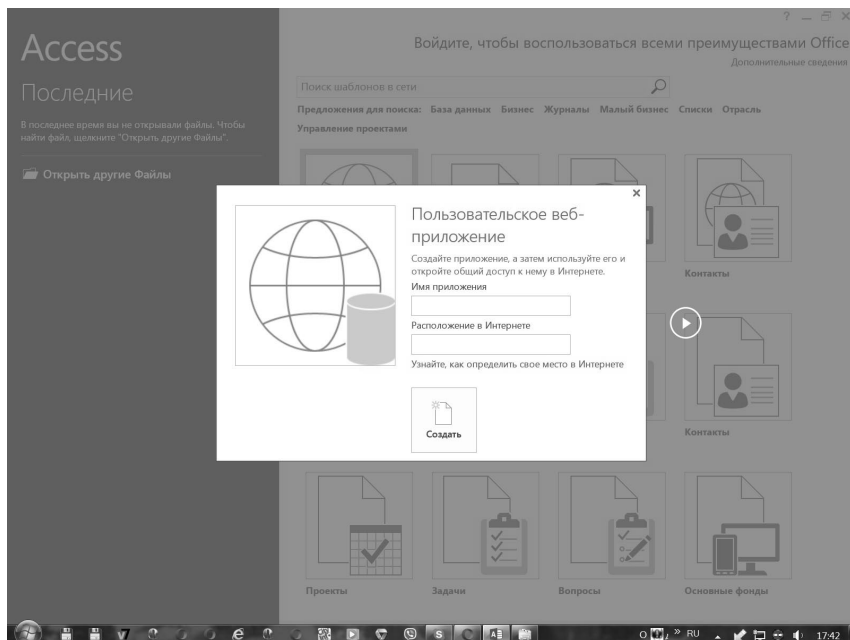


Рис. 2. Окно Access 2013 для задания имени и расположения web-приложения в Интернете

менно с созданием таблиц создаются формы для представления данных таблицы в браузере – одна в виде списка, а другая в виде таблицы.

Для использования разработанного в Access 2013 web-приложения достаточно иметь браузер и подключение к Интернету и необходимо зарегистрироваться на сайте SharePoint.

После открытия web-приложения в браузере возможны через формы просмотр, фильтрация, добавление, удаление, редактирование записей таблиц согласно правам, назначаемым с помощью трех уровней разрешений SharePoint: Разработчик, Автор, Читатель. Разработчик имеет полный доступ и может изменять объекты веб-приложения. Автор может изменять данные, но не может изменять структуру БД, а Читатель может только читать существующие данные.

В БД web-приложения, как и в локальной БД, с помощью запросов на выборку можно сформировать пользовательское представление о данных. В запросах, так же как и в запро-

сах к локальной БД, могут содержаться условия отбора, вычисляемые поля, выполняться группировка записей, использоваться параметры запроса. Запросы создаются в режиме Конструктора.

Для реализации более сложной логики приложения, автоматизации выполнения задач, связи различных объектов БД, добавления функциональных возможностей можно использовать доступный при разработке web-приложения язык макросов. В web-приложении могут быть созданы макросы трех типов: внедренные макросы, изолированные макросы, макросы данных.

Такая возможность работы с многопользовательской БД не требует от пользователей приобретения специальных навыков работы с БД.

Программное обеспечение технологии работы с многопользовательской БД развивается динамично, что требует постоянного совершенствования методического подхода к обучению этой технологии.

Литература

1. Бекаревич, Ю.Б. Самоучитель Microsoft Access 2013 / Ю.Б. Бекаревич, Н.В. Пушкина. – СПб.: БХВ-Петербург, 2014. – 464 с.

2. Оскерко, В.С. Практикум по технологиям баз данных: учеб. пособие с грифом Министерства образования РБ / В.С. Оскерко, З.В. Пунчик. – Минск: БГЭУ, 2004. – 170 с.

3. Оскерко, В.С. Технологии баз данных и знаний: учеб. пособие с грифом Министерства образования РБ / В.С. Оскерко, З.В. Пунчик. – Минск: БГЭУ, 2015. – 215 с.

Статья поступила 29.01.2016



С.В. Корженевич, учитель начальных классов ГУО «Гимназия №1 имени Ф.Я.-Перца г. Пинска»,

П.И. Личевский, учитель белорусского языка и литературы ГУО «Гимназия №1 имени Ф.Я.Перца г. Пинска»

Использование программы Plickers в образовательном процессе

С давних времен основным учебным ресурсом учащихся был и остается обычный полиграфический учебник. Список остальных образовательных ресурсов, доступных обычной школе, был достаточно коротким.

В эпоху информатизации, характеризующейся применением средств информационно-коммуникационных технологий во многих сферах деятельности человека, в том числе в сфере образования, мы имеем уникальную возможность рационального сочетания традиционных образовательных ресурсов с современными медиа-ресурсами. «Медиа-ресурс – это информационный ресурс, предназначенный для специфического восприятия через отдельный сенсорный канал или их совокупность» [1].

Под образовательными медиа-ресурсами понимают «совокупность всех электронных и компьютерных мультимедиа средств, предназначенных для обучения или поддержки обучения» [2]. Примером образовательного медиа-ресурса могут служить программы для контроля уровня предметных знаний и умений учащихся (в частности, программа Plickers). Главным достоинством этой программы является интерактивность, т.е. взаимодействие с пользователем.

Значение программы Plickers заключается и в том, что в течение нескольких минут можно определить уровень сформированности предметных знаний и умений учащихся с целью последующей их незамедлительной корректировки. Учителю достаточно разработать тест по интересующей его теме, указать правильные варианты ответов, а затем разместить этот тест в специальной закладке на сайте <https://plickers.com> и научиться пользоваться выше указанной программой.

Поэтому предлагаем алгоритм работы с этой программой. Он имеет следующий вид:

1. Зайти на сайт <https://plickers.com>.
2. Зарегистрироваться (для этого открыть закладку *Sign Up*; ввести имя, фамилию, логин и пароль).
3. Открыть закладку *Classes* и создать класс: написать название класса (можно выбрать год обучения, предмет, цвет).
4. Там же создать список учащихся.
5. Открыть закладку *Cards*, выбрать вариант карточки и распечатать.
6. Перейти на закладку *Library*: создать папку, дать название, открыть папку и внести тест. Каждый вопрос теста добавить в очередь.
7. На смартфон либо планшет установить с помощью Google Play программу Plickers.
8. На компьютере открыть закладку *Live View*.
9. На смартфоне либо планшете запустить программу Plickers. Там появится название класса (классов). Нажать на название класса – появятся тестовые вопросы. Нажать на первый вопрос. Этот же вопрос и список учеников класса можно будет увидеть на экране компьютера.
10. Навести фотокамеру смартфона на класс, попросить учеников поднять карточки соответствующим образом. Нажать на значок фотокамеры, обвести фотокамерой класс.
11. После ответов учеников на все вопросы, перейти на закладку *Reports*, открыть ее и проанализировать ответы.

Педагоги нашей гимназии первыми в городе овладели программой Plickers и активно используют ее в своей практической деятельности как на учебных занятиях для осуществления поурочного и тематического контроля, так и во внеклассной работе.

Систематическое использование данной программы способствует получению оперативной обратной связи. При помощи этой программы можно легко и быстро определить, какие предметные знания и умения усвоены учениками хорошо, а какие еще необходимо повторять и отрабатывать. Перед учителем открывается возможность быстро организовать коррекционную работу по ликвидации пробелов у учащихся.

Литература

1. Мультимедийные интерактивные медиаресурсы в образовательном процессе: реалии и перспективы развития / Ф.О. Каспаринский // [Электронный ресурс]. – Режим доступа: <http://istina.msu.ru/media/publications/articles/65b/1a3/1760498/12BioEdu06.pdf> – Дата доступа: 15.01.2016.

2. Основы разработки электронных образовательных ресурсов / [Электронный ресурс]. – Режим доступа: <http://kito.bspu.by/adminpanel/vendor/kcfinder/upload/files/lekzii>. – Дата доступа: 11.02.2016.

Статья поступила 05.06.2016



*А.А. Шелестович, учитель информатики
Государственного учреждения образова-
ния «Озерковская средняя школа», Грод-
ненская область, Мостовский р-н*

Kodi Game Lab как способ развития базовых основ программирования в игровой форме у школьников младшего и среднего возраста

Поиск новых форм и приемов обучения школьников в наше время – явление не только закономерное, но и необходимое. И это понятно: в свободной школе, к которой мы идем, каждый не только сможет, но и должен работать так, чтобы использовать все возможности собственной личности. На уроках особое место занимают такие формы занятий, которые обеспечивают активное участие в уроке каждого ученика, повышают авторитет знаний и индивидуальную ответственность, заинтересованность в результате своего труда. Эти задачи можно успешно решать через технологию игровых методов обучения. (Значение игры невозможно исчерпать и оценить развлекательными возможностями. В том и состоит ее феномен, что игра, являясь развлечением, формой отдыха, способна перерасти в обучение, творчество. [1, С.61])

Сегодня время программистов. Дело в том, что мы все программисты. Мы программируем стиральные машинки, хлебопечки и кофеварки, мы программируем будильники и календари в мобильных. Мы управляем движением такси и развозчиков пиццы, когда заказываем их услуги с помощью мобильных приложений. Однако в этом случае способ программирования настолько прост, что мы, не задумываясь, сдаем экзамен на использование сложнейших автоматических устройств и программных продуктов. Мы выполняем сложнейшие операции на Excel, становимся профессиональными бухгалтерами с программой 1С, делаем презентации в Power Point, рисуем в Photoshop и так далее. Думаю, даже самый отсталый пользователь компьютера умеет пользоваться не менее, чем пятью различными программными продуктами. И при этом нам кажется, что программирование – это

очень сложно, не для простых умов, никогда не понять, необходимо специальное образование и прочее. Значит, программирование нужно не только «избранным». Общеизвестно, что все люди играют в игры, только не все любят в этом признаваться. Игра – самое серьезное и глубокомысленное занятие ребенка. Более того, игра для ребенка – естественное состояние. Существует мнение, что игры не должны занимать ведущую роль в учебном процессе потому, что учеба – это подготовка к жизни, а жизнь – серьезная штука. На самом деле для ребенка нет более серьезного занятия, чем игра. Именно в играх человек познает мир, вырабатывает свой кодекс чести и т.д. Объявляя свою жизнь серьезной и недоступной для игр, мы сами делаем ее скучной и для себя, и для детей. Часто мы, беспокоясь за носы своих детей, твердим им, чтобы смотрели под ноги, не замечая, как они перестают смотреть на звезды. Как известно, ведущей деятельностью ребенка младшего и среднего школьного возраста является игровая, на этом и основываются следующие виды взаимодействия ребенка с компьютерными технологиями. Считается, что компьютерные игры – один из лучших способов тренировки логического мышления. Они всегда логичны, последовательны и при этом увлекательны. Кроме того, компьютерные игры начинаются с тренировки простых навыков, постепенно переходя на более сложные уровни, без травмирующей любого человека, а тем более ребенка, оценки результатов.

Но игры не только тренируют логическое мышление, они также помогают осваивать навыки программирования. Процесс игры полностью моделирует процесс программирования. Это игры, а не обучающие программы. Создано множество интересных «программерских» игр, которые можно вводить уже с начальных классов. Наверное, каждый согласился бы изучить основы программирования в игровой форме. В принципе, курс начального обучения программированию может строиться на основе любого исполнителя, а также нескольких исполнителей. На сегодняшний день в мире появилось очень много детских языков программирования. Препятствием использования подобных ресурсов может быть и неосведомленность об их существовании, и незнание языка. Действительно англоязычных курсов (Alice, Scratch, Code.org, Minecraft Blockly, Microsoft Imagine, JavaRush, Code Combat, Codecademy, Ruby Warrior, Kodu Game Lab и т.д.) и игр зна-

чительно больше, но существуют и русскоязычные игры, обучающие программированию (ПиктоМир, Colobot, Hexlet).

Остановимся на более современном продукте – сервисе Kodu. Сервис позиционируется как среда для разработки игр. Kodu представляет собой визуальную среду для создания казуальных игр без программирования для PC и игровой консоли Xbox 360, которая ориентирована на детскую и подростковую аудиторию. Для описания всех команд используются интуитивно понятные иконки, на которых показано, для выполнения какого действия они предназначены. Kodu разрабатывалась совместно с исследовательскими университетами США, чтобы предложить школьникам и начинающим программистам возможность забавного и увлекательного создания виртуальных миров. Дети могут развлекаться и развиваться одновременно: изучать математику, геометрию и прочие важные научные аспекты [3]. На Kodu достаточно просто писать несложные игры: лабиринты, гонки, простые аркады или стрелялки. Может быть, это и не совсем программирование в его классическом понимании. Но при помощи Kodu ребенку можно показать, что компьютер можно и нужно использовать не только для потребления того, что создано другими, но и для создания чего-то своего: уникального и неповторимого. А там и до более серьезных вещей уже недалеко. Главная часть Kodu – программный пользовательский интерфейс. Программирование происходит на основе графики, путем переключения различных экранов и кликания по различным иконкам. Вид среды состоит из страниц с правилами и условиями действий. Создание игр происходит исключительно визуально. Вы создаете поведение игрока, управляете визуальным рядом, звуками и сценариями. Kodu прост и интуитивно понятен. Он предоставляет вам все возможности для реализации творческого мышления ученика в создании игровых миров. Творческий процесс создания игрушек заключается в том, что ученику дается графический пользовательский интерфейс с набором действий и контентом, а вы уже на свое усмотрение строите из этого игровые сцены. Тем не менее, drag-and-drop программирование используется с так называемым языком высокого уровня Kodu Language. С его помощью можно управлять параметрами и настройками игрового мира, коллизиями, цветом и визуализацией. В нем имеется редактор ландшафта и миров произвольных формы и размера, редактор

игрового поведения, 20 различных персонажей с различными способностями [3]. Манипулируя исполнителями в их виртуальных мирах, обучающиеся получают первоначальный опыт работы с базовыми алгоритмическими структурами: при программировании в Kodu выбираются визуальные фрагменты для условия (WHEN) и действия (DO), т.е. осуществляется пропедевтика базовых алгоритмических конструкций. Преимуществом среды Kodu является возможность визуализации написанного сценария, что позволяет установить соответствие между планируемыми действиями исполнителя и его реальным поведением. В Kodu уже есть готовые игры, поэтому можно не делать игры, а просто играть в них, или редактировать. Через меню можно за-гружать игры. Есть возможность публиковать свои игры на специальном сервисе, чтобы в них смогли поиграть другие пользователи. Используя сервис Kodu, учитель получает уникальную среду, в которой обучающийся имеет возможность осознать сущность и природу таких базовых понятий информатики, как «алгоритм», «исполнитель», «программа», «подпрограмма», «модель». Обучающиеся осваивают процесс управления исполнителем на основе системы команд этого исполнителя, получают опыт моделирования среды и деятельность исполнителя в зависимости от условий, отношений между объектами; осваивают один из основных подходов оптимизации структуры программы – написание и вызов подпрограмм (вспомогательных алгоритмов). Так, в ходе разработки сценария игры и его практической реализации обучающиеся учатся управлять своей деятельностью, контролировать ее и вносить свои коррективы, проявлять инициативность и самостоятельность; создаются естественные условия для формирования умений понимать причины успеха/неуспеха учебной деятельности и способности конструктивно действовать даже в ситуациях неуспеха. Именно работа над созданием продукта способствует формированию навыков решения проблем творческого и поискового характера, планирования учебных действий в соответствии с поставленной задачей. И именно в ходе программирования обучающийся учится оценивать эффективность способов достижения результата, выбирать оптимальный вариант и аргументировать свой выбор. Указанные аспекты реализации дидактического потенциала визуальной среды программирования Kodu позволяют сделать вывод, что внедрение данной среды в образовательный процесс со-

здает объективные условия для ранней профилизации обучающихся, формирования мотивации у обучающихся к получению компьютерного образования через получение практического опыта. Учащиеся могут научиться не только программированию, но и сотрудничеству, логике и творчеству.

Литература

1. Kodu Game Lab [Электронный ресурс]. – Режим доступа: <http://www.kodugamelab.com/>. – Дата доступа: 03.06.2016.

2. Создаем 3D игры вместе с Kodu Game Lab [Электронный ресурс]. – Режим доступа: <http://www.microsoftvirtualacademy.com/training-courses/games-creating-with-kodu-game-lab-rus>. – Дата доступа: 03.06.2016.

Статья поступила 02.06.2016



САММИТ ТЕКНОЛОДЖИЗ

САММИТ
САММИТ
САММИТ
САММИТ
САММИТ
САММИТ
САММИТ
САММИТ
САММИТ
САММИТ
САММИТ

- Вся продукция и производство сертифицированы
- Производство компьютеров и мониторов Summit Systems®
- Мощные сервера
- Компьютерные классы
- Проектирование и монтаж ЛВС
- Гарантийное обслуживание и сервисная поддержка на всей территории Беларуси
- Прямые связи с ведущими производителями:
Intel, GigaByte, Microsoft, АОС и др.

Компьютерный класс -

из классных компьютеров!

Любая конфигурация под заказ

Материнские платы GigaByte, Intel

CPU Intel, AMD

Мониторы Summit Systems®, АОС и др.

Модернизация и ремонт компьютеров и мониторов

Выезд специалиста для консультации

Консалтинг и информационная поддержка

СООО «САММИТ ТЕКНОЛОДЖИЗ»

УНН 600259287

ул. Кульман 1, корп.3, этаж 3.

220013, г. Минск, Республика Беларусь

тел: +375 (17) 237-35-70; 237-35-71; 237-35-72

факс: +375 (17) 217-73-38

e-mail: summit@summit.by

www.summit.by

Сервисный центр: тел: +375 (17) 234-27-72

Региональные сервисные представители:

г. Брест: +375 (162) 46-02-83;

г. Витебск: +375 (212) 36-61-65;

г. Гомель: +375 (232) 40-67-90;

г. Гродно: +375 (29) 134-49-15;

г. Могилев: +375 (222) 25-03-26; 31-17-52

S U M M I T
S Y S T E M S



НАУЧНЫЕ ПУБЛИКАЦИИ

Н.И. Листопад, д.т.н., профессор, заведующий кафедрой информационных радиотехнологий Белорусского государственного университета информатики и радиоэлектроники,

И.А. Карук, магистрант Белорусского государственного университета информатики и радиоэлектроники,

А.А. Хайдер, аспирант Белорусского государственного университета информатики и радиоэлектроники

Алгоритмы поиска кратчайшего пути и их модификация

В статье представлен обзор основных тенденций и методов поиска кратчайшего пути передачи информации в сетях телекоммуникаций. Описаны основные алгоритмы и их модификация. Особое внимание уделено модифицированному алгоритму Дейкстры, учитывающему при поиске кратчайшего пути требования QoS. Описан сам алгоритм и представлена его диаграмма классов.

Введение

Мультисервисная сеть должна обладать надежностью и обеспечивать высокую скорость и низкую стоимость передачи данных в расчете на единицу объема информации.

Основная задача мультисервисных сетей нового поколения заключается в обеспечении работы разнородных информацион-

ных и телекоммуникационных систем и приложений в единой транспортной среде, когда для передачи данных и мультимедийного трафика используется единая инфраструктура.

Сеть нового поколения отличается новыми возможностями:

а) универсальным характером обслуживания разных приложений;

б) независимостью от технологий услуг связи и гибкостью получения набора, объема и качества услуг;

в) полной прозрачностью взаимоотношений между поставщиком услуг и пользователями.

Интеграция трафика разнородных данных и речи позволяет добиться качественного повышения эффективности информационной поддержки управления предприятием или организацией, при этом использование интегрированной транспортной среды позволяет снизить издержки на создание и эксплуатацию сети. Оптимально функционирующая мультисервисная сеть использует единый канал для передачи данных разных типов, позволяет уменьшить разнообразие типов оборудования, применять единые стандарты, технологии и централизованно управлять коммуникационной средой.

Базовыми понятиями мультисервисных сетей являются обеспечение требований *QoS (Quality of Service)* и *SLA (Service Level Agreement)*, то есть качество обслуживания и соглашение об уровне (качестве) предоставления услуг сети. Переход к новым мультисервисным технологиям изменяет саму концепцию предоставления услуг, когда качество гарантируется не только на уровне договорных соглашений с поставщиком услуг и требований соблюдения стандартов, но и на уровне технологий и операторских сетей.

Средства поддержки качества обслуживания в современных протоколах маршрутизации в последнее время достаточно сильно изменились, прежде всего, за счет пересмотра метрик, используемых при выборе маршрута. Во-первых, происходит отказ от топологических метрик – числа переприемов (hops), характерного для протокола RIP, и осуществляется переход на QoS-метрики, основанные на учете основных показателей качества обслуживания: скорости передачи, средних задержек, вариации задержек, уровня потерь в трактах передачи сети. В результате маршруты передачи пакетов того или иного трафика прокладываются с учетом QoS-показателей вдоль них. Во-вторых, все больше протоколов поддержи-

вают так называемые композитные (комбинированные) метрики, в рамках которых учитываются одновременно несколько основных QoS-показателей.

1. Алгоритмы поиска кратчайших путей

Благодаря своему широкому применению, теория о нахождении кратчайших путей в последнее время интенсивно развивается и используется в различных сферах деятельности, например, для нахождения оптимального маршрута между двумя объектами на местности (кратчайший путь от дома до университета), для нахождения оптимального маршрута при перевозках, в системах автопилота, в системах коммутации информационных пакетов в сети Internet и т.п.

Кратчайший путь можно определить, используя графокомбинаторные и потоковые математические модели. Приведем пример некоторых алгоритмов нахождения кратчайшего пути, базирующихся на теории графов:

- алгоритм Дейкстры (используется для нахождения оптимального маршрута между двумя вершинами);
- алгоритм Флойда (используется для нахождения оптимального маршрута между всеми парами вершин);
- алгоритм Беллмана-Форда (для нахождения кратчайшего пути от одной вершины графа до всех остальных);
- алгоритм A* (используется для нахождения маршрута с наименьшей стоимостью от одной вершины (начальной) к другой (конечной), используя алгоритм поиска по первому наилучшему совпадению на графе);
- алгоритм Джонсона (используется для нахождения кратчайшего пути между всеми парами вершин взвешенного ориентированного графа);
- волновой алгоритм (переборный алгоритм, основанный на методе поиска в ширину) находит путь между вершинами графа, содержащего минимальное количество промежуточных ребер);
- алгоритм Габова (используется для нахождения кратчайшего пути с помощью масштабирования);
- алгоритм Карпа (используется для отыскания цикла с наименьшим общим весом).

Указанные алгоритмы легко выполняются при малом количестве вершин в графе. При увеличении их количества задача поиска кратчайшего пути усложняется. В этой связи ак-

туальной является задача разработки собственных алгоритмов поиска оптимального пути или модификация существующих подходов на базе комбинированных метрик.

Рассмотрим более подробно вышеперечисленные существующие алгоритмы поиска кратчайшего пути.

Алгоритм Дейкстры

Алгоритм Дейкстры (*Dijkstra's algorithm*) [1, 2] – алгоритм на графах, изобретенный нидерландским ученым Э. Дейкстрой в 1959 году. Алгоритм позволяет находить кратчайшее расстояние от одной из вершин графа до всех остальных. Алгоритм работает только для графов без ребер отрицательного веса и широко применяется в программировании и технологиях, например, его использует протокол OSPF для устранения кольцевых маршрутов [1, 2]. Алгоритм Дейкстры решает задачу о кратчайших путях из одной вершины для взвешенного ориентированного графа $G = (V, E)$ с исходной вершиной s , в котором веса всех ребер неотрицательны ($\omega(u, v) \geq 0$ для всех $(u, v) \in E$), где V – это непустое множество вершин или узлов; E – это множество пар (в случае неориентированного графа неупорядоченных) вершин, называемых ребрами.

В процессе работы алгоритм Дейкстры поддерживает множество $S \subseteq V$, состоящее из вершин v , для которых $\delta(s, v)$ уже найдено (т.е. $d[v] = \delta(s, v)$). Алгоритм выбирает вершину

$u \in V \setminus S$ с наименьшим $d[u]$, добавляет u к множеству S и производит релаксацию всех ребер, выходящих из u , после чего цикл повторяется. Вершины, не лежащие в S , хранятся в очереди Q с приоритетами, определяемыми значениями функции d . Предполагается, что граф задан с помощью списков смежных вершин. Каждой вершине из V сопоставляется метка – минимальное известное расстояние от этой вершины до α . Алгоритм работает пошагово – на каждом шаге он «посещает» одну вершину и пытается уменьшать метки. Работа алгоритма завершается, когда все вершины посещены.

Алгоритм Флойда-Уоршелла

Алгоритм Флойда-Уоршелла – динамический алгоритм для нахождения кратчайших расстояний между всеми верши-

нами взвешенного ориентированного графа. Разработан в 1962 году Робертом Флойдом и Стивеном Уоршеллом [2, 3].

Краткое описание алгоритма. Пусть вершины графа $G = (V, E)$, $V = n$ пронумерованы от 1 до n и введено обозначение d_{ij}^k для длины кратчайшего пути от i до j , который кроме самих вершин i, j , проходит только через вершины $1 \dots k$. Очевидно, что d_{ij}^0 – длина (вес) ребра (i, j) , если таковое существует (в противном случае его длина может быть обозначена как ∞).

Существует два варианта значений d_{ij}^k , $k \in (1, \dots, n)$:

1. Кратчайший путь между i, j не проходит через вершину k , тогда $d_{ij}^k = d_{ij}^{k-1}$.

2. Существует более короткий путь между i, j , проходящий через k , тогда он вначале идет от i до k , а потом от k до j . В этом случае, очевидно, $d_{ij}^k = d_{ij}^{k-1} + d_{kj}^{k-1}$.

Таким образом, для нахождения значения функции достаточно выбрать минимум из двух обозначенных значений.

Тогда рекуррентная формула для d_{ij}^k имеет вид:

d_{ij}^0 – длина ребра (i, j) ;

$$d_{ij}^k = \min(d_{ij}^{k-1}, d_{ik}^{k-1} + d_{kj}^{k-1}).$$

Алгоритм Флойда-Уоршелла последовательно вычисляет все значения d_{ij}^k , $\forall i, j$ для k от 1 до n . Полученные значения d_{ij}^n являются длинами кратчайших путей между вершинами i, j .

Алгоритм Беллмана-Форда

Предложен независимо Ричардом Беллманом и Лестером Фордом [4]. Алгоритм Беллмана-Форда – алгоритм поиска кратчайшего пути во взвешенном графе. Алгоритм находит кратчайшие пути от одной вершины графа до всех остальных. В отличие от алгоритма Дейкстры, алгоритм Беллмана-Форда допускает ребра с отрицательным весом.

Опишем алгоритм по шагам.

Шаг 0. Пронумеруем все вершины из $G = (A, B)$ так, что $A = \{1, 2, \dots, p\}$ и при этом номер «1» имеет именно та вершина, из которой будут найдены кратчайшие пути во все остальные вершины. Построим далее матрицу $M = (m_{ij})$, $i, j = 1, 2, \dots, p$, положив

$$m_{ij} = \begin{cases} f((i, j)), & \text{если } (i, j) \in B \\ \text{клетка остается незаполненной,} & \text{если } (i, j) \notin B. \end{cases}$$

Шаг 1. Около первой строки матрицы M (слева от матрицы), поставим числовую пометку «0» и такую же пометку поставим над первым столбцом матрицы. Затем просмотрим помеченную строку слева направо и всякий раз, встречая клетку с числом, прибавим это число к метке строки и сумму поставим над столбцом, в котором эта клетка находится. Затем отразим метки столбцов относительно главной диагонали. Возникнут помеченные строки. С каждой из помеченных строк сделаем то же самое: просмотрим помеченную строку слева направо и всякий раз, встречая клетку с числом, прибавим это число к метке строки и сумму поставим в качестве метки над столбцом, в котором эта клетка стоит. При этом будем соблюдать принцип: «имеющуюся метку не менять». Затем метки столбцов отразим относительно главной диагонали и с помеченными строками вновь сделаем то же самое. И так далее, пока не окажутся помеченными все строки и все столбцы.

Шаг 2. Просмотрим строки таблицы в порядке возрастания их номеров. В каждой строке просматриваются клетки слева направо и всякий раз, когда встречается число, оно складывается с пометкой строки и сумма сравнивается с пометкой столбца, в котором найденное число расположено. Если сумма оказалась меньше, чем пометка столбца, то эта пометка столбца заменяется на упомянутую сумму. Если же сумма оказалась больше или равной пометке, то ничего не меняется. После такого просмотра всех строк новые пометки столбцов отражаются относительно главной диагонали и вся процедура повторяется. И так до тех пор, пока не будут прекращены какие бы то ни было изменения в пометках.

Шаг 3. Теперь по пометкам можно построить кратчайшие пути из первой вершины во все остальные. Фиксируем про-

извольную вершину k ($k = 2, 3, \dots, p$) и опишем кратчайший путь из первой вершины в вершину k . Во-первых, длина этого кратчайшего пути равна пометке λ_k , стоящей над столбцом номер k . Во-вторых, предпоследняя вершина в кратчайшем пути из первой вершины в вершину k находится так: в столбце номер k отыскиваем число, сумма которого с пометкой строки, в которой оно расположено, равна λ_k . Пусть номер строки, в которой оказалось найденное число, равен l . Тогда предпоследней вершиной в кратчайшем пути из 1 в k будет вершина l . Вершину, которая предшествует вершине l , надо отыскивать как предпоследнюю в кратчайшем пути из 1 в l и так далее.

Алгоритм A*

Алгоритм A* используется для нахождения маршрута с наименьшей стоимостью от одной вершины (начальной) к другой (конечной), используя алгоритм поиска по первому наилучшему совпадению на графе. В нем применяется оценка узлов, объединяющая в себе $g(n)$ – стоимость достижения данного узла, и $h(n)$ – стоимость прохождения от данного узла до цели:

$$f(n) = g(n) + h(n)$$

Поскольку функция $g(n)$ позволяет определить стоимость пути от начального узла до узла n , а функция $h(n)$ определяет оценку стоимости наименее дорогостоящего пути от узла n до цели, то справедлива следующая формула:

$$f(n) = \text{оценка стоимости наименее дорогостоящего пути решения, проходящего через узел } n.$$

Таким образом, при осуществлении попытки найти наименее дорогостоящее решение вначале необходимо попытаться проверить узел с наименьшим значением: если эвристическая функция $h(n)$ удовлетворяет некоторым условиям, то поиск A* становится и полным, и оптимальным.

Анализ оптимальности поиска A* является несложным, если этот метод используется в сочетании с алгоритмом Tree-Search. В таком случае поиск A* является оптимальным, при условии, что $h(n)$ представляет собой допустимую эвристическую функцию, т.е. при условии, что никогда не переоценивает стоимость достижения цели. А поскольку $g(n)$ – точная стоимость достижения узла n , из этого непосредственно

следует, что функция $f(n)$ никогда не переоценивает истинную стоимость достижения решения через узел n .

Одно заключительное наблюдение состоит в том, что среди оптимальных алгоритмов такого типа (алгоритмов, которые разворачивают пути поиска от корня) поиск A^* является оптимально эффективным для любой конкретной эвристической функции. Это означает, что не гарантируется развертывание меньшего количества узлов, чем в поиске A^* , с помощью какого-либо иного оптимального алгоритма.

Те соображения, что поиск A^* , как один из всех подобных алгоритмов, является действительно полным оптимальным и оптимально эффективным, не означают, что поиск A^* может служить ответом на все потребности в поиске. Сложность заключается в том, что при решении большинства задач количество узлов в пределах целевого контура пространства состояний все еще зависит экспоненциально от длины решения. Экспоненциальный рост происходит, если ошибка эвристической функции растет не быстрее, по сравнению с логарифмом фактической стоимости пути. В математических обозначениях условие субэкспоненциального роста состоит в следующем:

$$|h(n) - h^*(n)| \leq o(\log h^*(n)),$$

где $h(n)$ – истинная стоимость достижения цели из узла n . Почти для всех практически применяемых эвристических функций эта ошибка, по меньшей мере, пропорциональна стоимости пути, и происходящий в связи с этим экспоненциальный рост в конечном итоге превосходит возможность любого компьютера. По этой причине на практике стремление находить оптимальное решение часто не оправдано. Иногда вместо этого целесообразно использовать варианты поиска A^* , позволяющие быстро находить неоптимальные решения, а в других случаях – разрабатывать эвристические функции, которые являются более точными, но не строго допустимыми.

Алгоритм Джонсона

Алгоритм Джонсона находит кратчайшие пути для всех пар вершин за время $O(V^2 \log V + VE)$ и основан на идее изменения весов (*reweighting*). Если веса ребер графа неотрицательны, то можно найти кратчайшие пути между всеми парами вершин, применив алгоритм Дейкстры к каждой

вершине. При этом данный алгоритм либо возвращает матрицу весов кратчайших путей, либо сообщает, что в графе имеется цикл отрицательного веса. Если же в графе имеются ребра с отрицательным весом, то можно попытаться свести задачу к случаю неотрицательных весов, заменив весовую функцию ω на новую функцию $\hat{\omega}$. При этом должны выполняться следующие свойства.

а) Кратчайшие пути не изменились: для любой пары вершин $u, v \in V$ кратчайший путь из u в v с точки зрения весовой функции ω является также кратчайшим путем с точки зрения $\hat{\omega}$ и наоборот.

б) Все новые веса $\hat{\omega}(u, v)$ неотрицательны.

Отсюда следует, что новая весовая функция $\hat{\omega}$ обладает теми же свойствами и может быть построена за время $O(VE)$ [6].

Алгоритм Габова

Рассмотрим взвешенный ориентированный граф $G = (V, E)$, в котором веса ребер являются целыми неотрицательными числами, не превосходящими W . Определим, как можно найти кратчайшие пути из одной вершины за время $O(E \log W)$.

Пусть $k = \lceil \log(W + 1) \rceil$ – количество битов в двоичном представлении числа W . Для $i = 1, 2, \dots, k$ положим $\omega_i(u, v) = \lfloor \omega(u, v) / 2^{k-i} \rfloor$ (иными словами, $\omega_i(u, v)$ получается из $\omega(u, v)$ отбрасыванием $k - i$ младших битов в двоичном представлении числа $\omega(u, v)$). Например, если $k = 5$ и $\omega(u, v) = 25 = \langle 11001 \rangle$, то из $\omega_3(u, v) = \langle 110 \rangle = 6$. В частности, ω_1 принимает только значения 0 и 1, определяемые старшим разрядом, а $\omega_k = \omega$.

Пусть $\delta(u, v)$ – вес кратчайшего пути из u в v относительно весовой функции ω_i (в частности, $\delta_k(u, v) = \delta(u, v)$). Алгоритм находит сначала все $\delta_1(s, v)$ (s – исходная вершина), а затем все $\delta_2(u, v)$, и так далее, пока не дойдет до $\delta_k(s, v) = \delta(s, v)$. Далее, мы полагаем, что $|E| \geq |V| - 1$; стоимость нахождения δ_i при известном δ_{i-1} есть $O(E)$, так что алгоритм будет работать за время, равное $O(kE) = O(E \log W)$.

Такой план решения задачи – замена исходных данных их двоичными приближениями с последовательными уточнением – называется масштабированием (*scaling*) [6].

Алгоритм Карпа

Пусть $G = (V, E)$ – ориентированный граф с весовой функцией $\omega: E \rightarrow R$, и пусть $n = |V|$. Средним весом (*mean weight*) цикла $c = \langle e_1, e_2, \dots, e_k \rangle$, где e_j – ребра графа, назовем число

$$\mu(c) = \frac{1}{k} \sum_{i=1}^k \omega(e_i).$$

Пусть $\mu^* = \min \mu(c)$, где c пробегает все (ориентированные) циклы. Не ограничивая общности (поскольку можно добавить дополнительную начальную вершину), будем считать, что каждая вершина $v \in V$ достижима из некоторой вершины s . Через $\delta(s, v)$ обозначим вес кратчайшего пути из s в v ; пусть $\delta_k(s, v)$ – вес кратчайшего пути из s в v , состоящего в точности из k ребер (если такого пути нет, полагаем $\delta_k(s, v) = \infty$) [6].

2. Модифицированный алгоритм Дейкстры

Рассмотрим более подробно модифицированный алгоритм Дейкстры как один из возможных алгоритмов поиска оптимального пути при комбинированной метрике, объединяющей в себя такие метрики, как пропускная способность канала связи, вероятность потерь пакетов, задержка в передаче пакетов и вариации задержки при минимальной стоимости передачи единицы информации. Модификация алгоритма подробно описана в [7] и заключается в отбрасывании в процессе поиска тех путей, на которых не выполняются ограничения требований (ограничений) по качеству обслуживания и новом способе описания и вычисления меток узлов.

Пусть путь p – множество ребер, соединяющих узел источника и узел получателя от источника s к получателю t [7]. В качестве QoS параметров каждого канала связи (ребра графа) будем рассматривать полосу пропускания Y_e , задержку D_e , вариации задержки J_e и вероятность потери пакетов Z_e . Для каждого из возможных путей из s в t будут справедливы следующие соотношения:

$$Y_{s,t} = \max_{e \in p} \{Y_e\}; D_{s,t} = \sum_{e \in p} D_e; J_{s,t} = \sum_{e \in p} J_e; Z_{s,t} = \prod_{e \in p} Z_e. \quad (1)$$

Ограничения на метрики, обеспечивающие требования заданного качества обслуживания, можно представить в следующем виде [7]:

$$Y_{s,t} - Y^{\min} \geq 0; D^{\max} - D_{s,t} \geq 0; J^{\max} - J_{s,t} \geq 0; X_{s,t} - X^{\min} \geq 0. \quad (2)$$

$$X_e = \ln(1 - Z_e) \quad (3)$$

В исходном алгоритме Дейкстры метки j -го узла, которого можно достичь из узла s через соседний узел i , имеют вид $[R_{s,j}, i]$, где величина стоимости $R_{s,j}$, соответствующая данному пути, аддитивна и вычисляется по формуле $R_{s,j} = R_{s,i} + R_{i,j}$, где $R_{i,j}$ – стоимость ребра e_{ij} , а величина $R_{s,i}$ берется из метки i -го узла. Вместо этого, введем метку, имеющую 6 компонентов: $[R_{s,j}, Y_{s,j}, D_{s,j}, J_{s,j}, X_{s,j}, i]$. Новую метку при переходе из узла i в j будем вычислять следующим образом:

$$D_{s,j} = D_{s,i} + D_{i,j} \quad (4)$$

$$J_{s,j} = J_{s,i} + J_{i,j} \quad (5)$$

$$X_{s,j} = X_{s,i} + X_{i,j} \quad (6)$$

$$Y_{s,j} = \min \{ Y_{s,i}, Y_{i,j} \} \quad (7)$$

Комбинированная метрика представлена в виде следующей свертки:

$$r = -w_Y \frac{Y_{s,j}}{Y^{\min}} + w_D \frac{D_{s,j}}{D^{\max}} + w_J \frac{J_{s,j}}{J^{\max}} + w_X \frac{X_{s,j}}{X^{\min}}, \quad (8)$$

$$R_{s,j} = \begin{cases} r, & \text{если для } D_{s,j}, J_{s,j}, X_{s,j}, Y_{s,j} \text{ выполняются условия (2)} \\ \infty, & \text{если для } D_{s,j}, J_{s,j}, X_{s,j}, Y_{s,j} \text{ не выполняется хотя бы одно из условий (2)} \end{cases} \quad (9)$$

Блок-схема алгоритма представлена на рис. 1. Более подробное описание алгоритма и назначение его блоков представлены в работе [7].

Программа для нахождения кратчайшего пути написана на языке C++. Для ее реализации использовались принципы объектно-ориентированного программирования: абстракция, инкапсуляция, наследование, полиморфизм, использование объектов и классов.

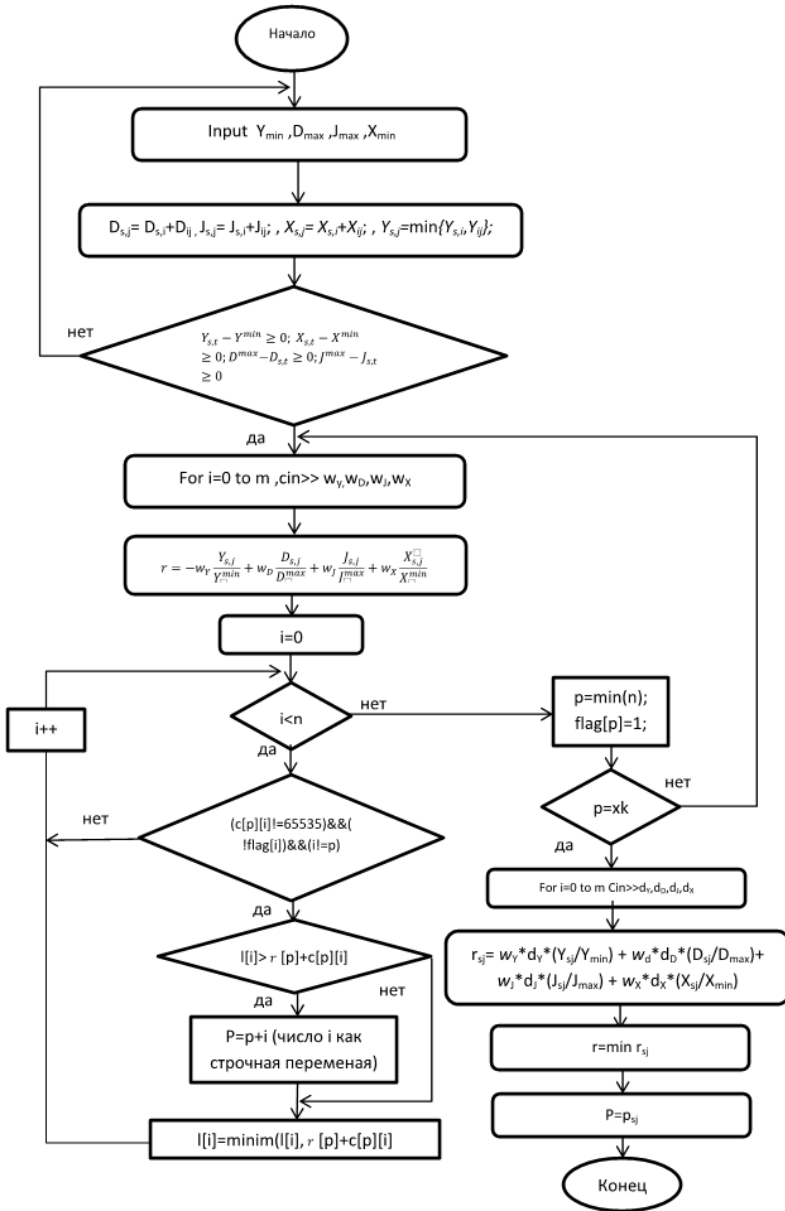


Рис. 1. Блок-схема алгоритма поиска оптимального пути

В качестве абстракции был использован класс весовых и стоимостных коэффициентов (`weightsFactors` и `costFactors`).

Пример инкапсуляции – это использование данных для нахождения пути: узел начала и конца, R-параметр, путь и др. Данные были сокрыты от ошибочного или специального изменения, так как это может повлиять на работу программы.

Для нахождения кратчайшего пути были созданы следующие специальные классы.

`boundaryConditions` (граничные условия) – класс, содержащий граничные условия, которые необходимы для отсеивания путей, не удовлетворяющих граничным условиям.

`CostFactors` (стоимостной коэффициент) – класс, содержащий стоимостные коэффициенты, необходимые для расчета пути с учетом стоимостных коэффициентов. Основные методы – это ввод и хранение данных.

`edge` (ребра) – класс, содержащий данные, входящие в ребро: задержка, пропускная способность, вероятность потери пакетов, вариация задержки, и r-параметр, в который рассчитываются все используемые параметры. Важным параметром является булевская переменная, которая сигнализирует о прохождении пути через узел. Основные методы – это ввод и хранение данных и установка флага прохождения пути.

`host` (узел) – класс, содержащий данные, входящие в ребро, которые суммируются по пути прохождения: задержка, пропускная способность, вероятность потери пакетов, вариация задержки, и r-параметр, в который рассчитываются все используемые параметры. Важным параметром является булевская переменная, которая сигнализирует о прохождении пути через узел. Номер узла, необходимый для нумерации пути, хотя как таковыми номера узлов используются лишь для нумерации найденного пути. Основные методы – это хранение данных и установка флага прохождения пути.

`way` (путь) – класс, содержащий номер начального и конечного узла, начальное значение r-параметра и его промежуточные значения при нахождении пути, номер узла в процессе нахождения пути. Основные методы – это хранение данных, сохранение пути, его очистка при следующем нахождении.

`weightsFactors` (весовой коэффициент) – класс, содержащий весовые коэффициенты, которые необходимы для расчета пути с учетом стоимостных коэффициентов. Основные методы – ввод и хранение данных.

Основные функции программы – это нахождение кратчайшего пути. Расчет происходит по двум модифицированным алгоритмам. Для хранения данных используются массивы, так как это наиболее удобный и быстрый способ хранения и использования большого массива данных. Максимальный размер графа не ограничен, но в качестве бесконечности используется 2^{16} , это соответствует типу `double` в языке C++. Сложность расчета увеличивается с количеством узлов, для каждого алгоритма расчет производится по своей формуле.

Диаграмма введенных классов представлена на рисунке 2. Проведенные расчеты показали, что модификации алгоритма не дают серьезного ухудшения в скорости и сложности расчета кратчайшего пути.

Заключение

Разработка программного обеспечения на основе модифицированного алгоритма Дейкстры основана на замене весового параметра сложной весовой функцией и учете граничных QoS-параметров; предполагает выбор языка программирования и подходящей для его реализации среды и позволяет использовать преимущества программной платформы Microsoft.NET, линейки Visual Studio Express и других ресурсов.

Разработанный с учетом отмеченных условий модифицированный алгоритм поиска на графе (на основе алгоритма Дейкстры) обладает преимуществом автоматического поиска оптимального пути посредством координированного учета всего массива различных параметров других мультимедийных сервисов.

Будучи интегрированным в систему мультисервисных сетей нового поколения, разработанное программное обеспечение позволяет увеличить скорости сетевого пользования и оптимизировать многокритериальный поиск оптимального пути с учетом требований заданного качества обслуживания.

Область применения разработанного программного обеспечения включает в себя такие сферы как создание экспертных систем, разработка баз данных, систем навигации и слежения, картографических сервисов, оптимальных протоколов маршрутизации информационных потоков.

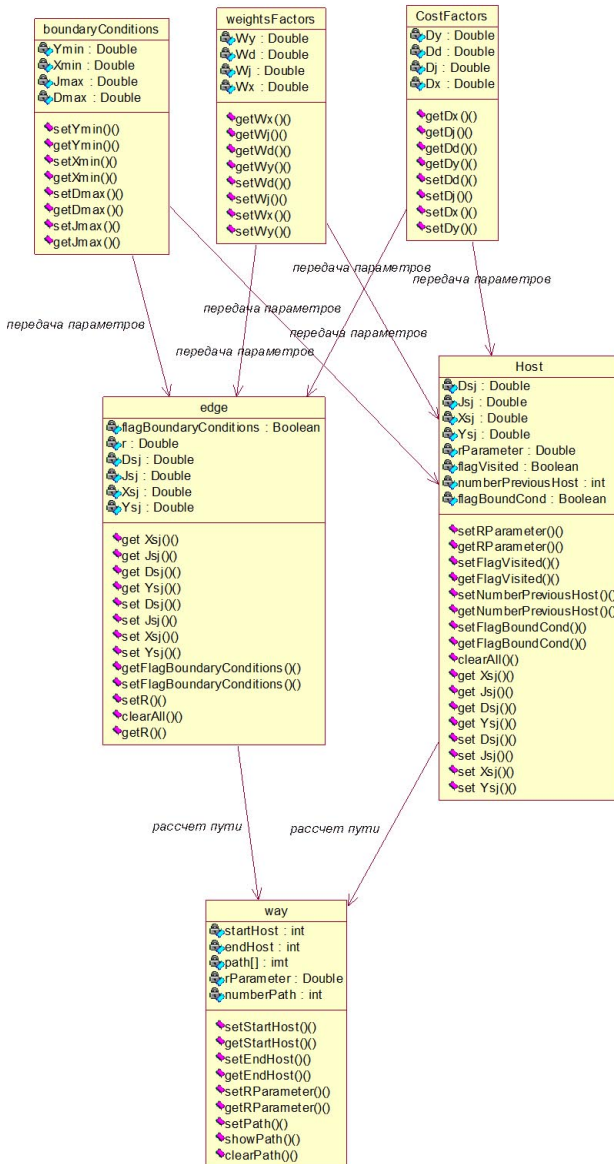


Рис. 2. Диаграмма классов модифицированного алгоритма Дейкстры

Литература

1. E.W. Dijkstra. A note on two problems in connexion with graphs. // Numerische Mathematik. V. 1 (1959), P.269-271.
2. Левитин А.В. // Алгоритмы: введение в разработку и анализ = Introduction to The Design and Analysis of Algorithms. – М.: Вильямс, 2006. – С.189-195, С.349-353.
3. Томас Х. Кормен, Чарльз И. Лейзерсон, Рональд Л. Ривест, Клиффорд Штайн Алгоритмы: построение и анализ = Introduction to Algorithms. – 2-е изд. – М.: Вильямс, 2006. – С.1296.
4. [Электронный ресурс]. – Режим доступа: <http://comp-science.narod.ru/KPG/BelmanFord.htm>.
5. Рассел С., Норвиг П. Искусственный интеллект: современный подход (AIMA) = Artificial Intelligence: A Modern Approach (AIMA). – 2-е изд. – М.: Вильямс, 2007. – 1424 с.
6. Томас Х. Кормен и др. Алгоритмы: построение и анализ. – 1-е изд. – М.: МЦНМО, 2004. – 523 с.
7. Н.И. Листопад, Ю.И. Воротницкий, А.А. Хайдер // Оптимальная маршрутизация в мультисервисных сетях телекоммуникаций на основе модифицированного алгоритма Дейкстры. // Вестник БГУ, серия 1. – 2015. – № 1. – С.70-76.

N.I. Listopad, I.A. Karuk, A.A. Hayder

Algorithms for Searching the Shortest Path and Its Modification

The article presents an overview of the main trends and methods of searching the shortest path information transmission in the telecommunications networks. The basic algorithms and their modifications are described. Special attention is given to the modified Dijkstra's algorithm that takes into account QoS requirements. The modified algorithm and its UML diagram are presented.

Статья поступила 02.05.2016



Е.М. Шевчик-Гирис, магистр педагогических наук, аспирант кафедры педагогики УО «Минский государственный лингвистический университет»

Полисемия термина «информационно-коммуникационная культура»

В статье рассматривается содержание понятия «информационно-коммуникационная культура», дается определение данного термина с позиций культурологии, педагогики, социологии, информатики, психологии и медицины, предлагается авторское определение этого понятия. Педагоги, социологи, культурологи, философы и представители медицины предложили множество определений понятия «информационно-коммуникационная культура», исходя из специфики научных исследований. Различные подходы в интерпретации данного понятия можно найти в работах таких исследователей как Кладова И.С., Паршенок Е.В., Матвеева Ю.А., Неумоина Е.Г., Антонова С.Г., Волков А.Н. и других.

Решением министров образования стран общеевропейского пространства от 14 мая 2015 г. Беларусь принята в Болонский процесс. Нашу страну этот шаг активно вовлекает в процесс глобализации и формирования единого информационного пространства и ставит сегодня перед системой образования важнейшие задачи по формированию информационно-коммуникационной культуры специалиста в условиях информационного общества.

Национальная стратегия устойчивого социально-экономического развития Беларуси на период до 2030 года (НСУР-2030), разработанная Научно-исследовательским экономическим институтом Минэкономики Республики Беларусь с привлечением министерств Республики Беларусь, а также международных специалистов от Всемирного банка, ПРООН, Глобального экологического фонда и экспертов из Германии, инициирует инновационную схему развития нашей республики, ориентированную на переход к постиндустриальному обществу и формированию информационного общества. «При-

оритетами государства станут инвестиции в развитие человеческого потенциала, формирование класса «интеллектуальных служащих» («людей знаний») через опережающее образование, культурно-духовную сферу, новое содержание труда и здоровый образ жизни» [7, С.18]. Государственные программы (Концепция информатизации системы образования Республики Беларусь до 2020 года и Государственная программа развития цифровой экономики и информационного общества на 2016-2020 годы) ориентированы на процесс информатизации образования и формирования современного информационного общества.

Особое внимание уделяется совершенствованию профессионального образования с ориентацией на подготовку специалиста, владеющего всеми необходимыми компетенциями (коммуникативными, профессиональными, информационно-технологическими), которые и составляют информационно-коммуникационную культуру личности.

Понятие «информационно-коммуникационная культура» активно используется отечественными и зарубежными учеными, является предметом исследования ученых из разных областей знаний: информатики, социологии, психологии, педагогики, культурологи, медицины и др. Терминологическая и содержательная стороны данного понятия рассматривалась нами с учетом исследований данной дефиниции учеными как в Республике Беларусь, так и за рубежом.

Впервые термин «информационная культура» был введен в 70-х гг. XX века в работах библиотекведов и библиографов. Первые работы с введением термина «информационная культура» нашли отражение в статьях К.М. Войханской и Б.Я. Смирновой «Библиотекари и читатели об информационной культуре» (1974г.) и Э.Л. Шапиро «О путях уменьшения неопределенности информационных запросов» (1975 г.). Авторы вкладывали в содержание понятия «информационная культура» умения и навыки читателей работать с литературой. В конце 80-х годов XX века данное словосочетание было введено в научную литературу Г.Г. Воробьевым «Твоя информационная культура» [3] и А.П. Сухановой «Информация и прогресс» [10]. Дальнейшей разработкой и дополнением понятия «информационно-коммуникационная культура» за-

нимались ученые в области социологии, информатики, педагогики, психологии, культурологии, медицины и других наук, конкретизируя и дополняя терминологическую и содержательную стороны данного понятия.

Так, ученые в области информатики (Неумоина Е.Г., Волков А.Н., Каракозов С.Д., Милитарев В.Ю. и др.) понимают под информационно-коммуникационной культурой компьютерную грамотность и умения личности работать с электронными сетями. Как отмечает Е.Г. Неумоина, «информационно-коммуникационная культура – это умение человека использовать информационные технологии для общения с людьми, получения, передачи и обработки информации» [8]. В свою очередь, А.Н. Волков считает, что «информационно-коммуникационная культура – это и хорошая осведомленность в области имеющихся программных приложений по функциональным задачам конкретных предметных областей, профессий и специальностей, программных приложений педагогической направленности (педагогических программных средств) и, в первую очередь, имеющихся в специализированных фондах программных приложений, признанных, имеющих высокий рейтинг, распространенных на рынке программного обеспечения, обладающих высокими показателями качества» [2].

Представители социологических наук (Каймин В.А., Смирнов Е.П., Суханов А.П., Кладова И.С., Матвеева Ю.А. и др.) придерживаются точки зрения, что главным критерием сформированности информационно-коммуникационной культуры личности является умение человека хорошо ориентироваться в постоянно меняющихся социальных условиях и способность самостоятельно и критически оценивать всю поступающую к нему информацию. В работах И.С. Кладовой информационно-коммуникационная культура рассматривается как «одна из составляющих общей культуры человека; совокупность его информационно-коммуникативной направленности и системы знаний и умений, обуславливающих творческую деятельность субъекта по оптимальному удовлетворению индивидуальных информационно-коммуникативных потребностей и задающих диалоговое взаимодействие в культурном пространстве» [4]. Социолог Ю.А. Матвеева понимает под информационно-ком-

муникационной культурой возможность одновременного выбора нескольких вариантов кодирования, восприятия и творческой переработки информации [5].

Что касается ученых в области культурологии (Антонова С.Г., Коган Е.А., Сулеймановой З.З. и др.), то их понимание «информационно-коммуникационной культуры» сводится к интерпретации данной культуры как способа функционирования человека в информационном обществе и как составной части общей культуры человека. Так, С.Г. Антонова подчеркивает связь информационно-коммуникационной культуры личности с духовной культурой личности и отмечает ее целостность, как целостна и сама духовная культура, отдельные аспекты которой привлекают внимание при изучении человека в различные периоды развития человечества [1].

Формирование информационно-коммуникационной культуры в педагогической системе наук, по мнению таких ученых как Гендина Н.И., Хеннер Е. К., Шестаков А. П., Паршенок Е.В., помогает достичь главной цели, которая стоит перед педагогикой сегодня – подготовить всестороннее развитую личность, способную жить и работать в условиях информационного общества.

По мнению Е.В. Паршенок, «информационно-коммуникационная культура» представляет собой «комплексное образование, обеспечивающее самостоятельный поиск необходимой информации, ее анализ и передачу, способность решать не только учебные, но и бытовые, профессиональные задачи с помощью современных средств коммуникации и информационно-коммуникационных технологий» [9].

С точки зрения психологии, «информационно-коммуникационная культура» подразумевает оптимальную реакцию личности на поступающую информацию и адекватное поведение личности в условиях недостаточного или избыточного информационного потока (Елистратова Н.Н., Стрельников А.В. и др.).

В медицине понятие «информационно-коммуникационной культуры» трактуется с позиций успешной коммуникации «доктор-пациент» и использования эффективных методов поиска информации медицинского характера (Мушников Д.Л., Поляков Б.А., Грязнова Е.В., Вайцехович Н.Ю.

и др.). «В структуре медицинской субкультуры как системе можно выделить следующие компоненты: гносеологический, организационно-праксеологический, коммуникативный, информационный, институциональный, нормативно-регулирующий, аксиологический. Коммуникативный компонент обеспечивает и регулирует культуру общения медицинских работников с пациентами и внутри самой медицинской субкультуры. Информационный компонент предполагает наличие, получение, необходимый уровень профессиональной информации и возможность обмена информацией субъектов медицинской деятельности» [6].

Исходя из анализа различных подходов исследователей к определению сущности понятия «информационно-коммуникационная культура», можно сделать вывод, что становление информационно-коммуникационной культуры представляет собой сложный и длительный процесс, который протекает одновременно с процессом обучения. Формирование информационно-коммуникационной культуры означает всестороннее гармоничное развитие личности в условиях информационного общества.

Таким образом, проанализировав трактовки понятия «информационно-коммуникационная культура», имеющие место в различных науках, мы пришли к выводу, что понятие *«информационно-коммуникационная культура личности»* – сложная многокомпонентная структура, представляющая собой систему знаний, умений и навыков по поиску, преобразованию и использованию информации, а также систему эффективных способов трансляций информационных ресурсов, обеспечивающее гармоничное функционирование личности в условиях информационного общества и является индикатором профессиональной, социальной и культурной компетентности личности.

Глубокий анализ трактовки термина «информационно-коммуникационная культура» с позиций различных наук способствует совершенствованию содержательной характеристики исследуемой нами дефиниции в контексте современного научного знания.

Сущностная характеристика понятия «информационно-коммуникационная культура»

Определение понятия	Автор (литературный источник)
Социология	
<p><i>Информационно-коммуникационная культура личности</i> может быть определена как одна из составляющих общей культуры человека; совокупность его информационно-коммуникативной направленности и системы знаний и умений, обуславливающих творческую деятельность субъекта по оптимальному удовлетворению индивидуальных информационно-коммуникативных потребностей и задающих диалоговое взаимодействие в культурном пространстве.</p>	<p>Кладова, И.С. Универсальность как условие формирования информационно-коммуникационной культуры личности / Кладова И.С. // Актуальные вопросы общественных наук: социология, политология, философия, история: сборник статей по материалам LVI международной научно-практической конференции (16 декабря 2015г.). – Новосибирск, 2015. – С. 42-49.</p>
<p>В основе <i>информационно-коммуникативной культуры</i> современного общества лежит возможность одновременного выбора нескольких вариантов кодирования, восприятия и творческой переработки информации</p>	<p>Матвеева, Ю.А. Информационно-коммуникативная культура как основная характеристика современного общества / Ю.А. Матвеева // Вестник Бурятского государственного университета. – 2010. – № 6. – С. 243-247.</p>
Психология	
<p><i>Информационно-коммуникационная культура</i> – способность индивида решать учебные, бытовые, профессиональные задачи с использованием информационных и коммуникационных технологий. ИКК проявляется, прежде всего, в деятельности при решении различных задач, которые могут быть выполнены с привлечением компьютера, средств телекоммуникаций, Internet и др.</p>	<p>Формирование учебной мотивации студентов через повышение информационно-коммуникативной культуры преподавателя / Методические рекомендации [Электронный ресурс]. – Режим доступа: http://ugkp.ru.../recomendacii_po_formirovaniyu_uchebnoy_motivacii. – Дата доступа: 20.02.2011.</p>

Медицина	
<p>В структуре медицинской субкультуры как системе можно выделить следующие компоненты: гносеологический, организационно-праксеологический, коммуникативный, информационный, институциональный, нормативно-регулирующий, аксиологический. Коммуникативный компонент обеспечивает и регулирует культуру общения медицинских работников с пациентами и внутри самой медицинской субкультуры. Информационный компонент предполагает наличие, получение, необходимый уровень профессиональной информации и возможность обмена информацией субъектов медицинской деятельности</p>	<p>Мясоедов, А.М. Медицинская субкультура: специфика, структура, динамика: автореф. дис. ... канд. культур.: 24.00.01 / А.М. Мясоедов. – М, 2012. – 29 с.</p>
Педагогика	
<p><i>Информационно-коммуникационная культура</i> представляет собой комплексное образование, обеспечивающее самостоятельный поиск необходимой информации, ее анализ и передачу, способность решать не только учебные, но и бытовые, профессиональные задачи с помощью современных средств коммуникации и информационно-коммуникационных технологий.</p>	<p>Паршонок, Е.В. Информационно-коммуникационная культура как составляющая социального опыта старшекласников с нарушениями зрения / Е.В. Паршонок // Специальное образование: опыт и перспективы развития: сб. мат-лов респуб. науч.-практ. конф., г. Могилев, 12 октября 2010 г. – Могилев: УО «МГУ им. А.А. Кулешова», 2011. – С. 106-110.</p>
Информатика	
<p><i>Информационно-коммуникационная культура</i> – это умение человека использовать информационные технологии для общения с людьми, получения, передачи и обработки информации.</p>	<p>Неумоина, Е.Г. Внедрение информационных технологий в учебный процесс / Е.Г. Неумоина [Электронный ресурс] – Режим доступа: http://it-education.ru/2011/section/78/3857/index.html. – Дата доступа: 21.02.1016.</p>

<p><i>Информационно-коммуникационная культура</i> – это и хорошая осведомленность в области имеющихся программных приложений по функциональным задачам конкретных предметных областей, профессий и специальностей, программных приложений педагогической направленности (педагогических программных средств) и, в первую очередь, имеющихся в специализированных фондах программных приложений, признанных, имеющих высокий рейтинг, распространенных на рынке программного обеспечения, обладающих высокими показателями качества</p>	<p>Волков А.Н. Новые информационно-коммуникационные технологии обучения профессии // Профессиональное образование. – 2001. – № 3.</p>
<p>Культурология</p>	
<p>Вместе с тем понятие <i>информационно-коммуникационной культуры</i> как бы подчеркивает связь ее с духовной культурой личности, а также целостность, как целостна и сама духовная культура, отдельные аспекты которой привлекают внимание при изучении человека в различные периоды развития человечества</p>	<p>Антонова, С.Г. Информационная культура личности. Вопросы формирования (в системе высшего образования) / С.Г.Антонова // Высшее образование в России. – 1994. – № 1. – С. 82-89.</p>

Литература

1. Антонова, С.Г. Информационная культура личности. Вопросы формирования (в системе высшего образования) / С.Г. Антонова // Высшее образование в России. – 1994. – № 1. – С.82-89.
2. Волков, А.Н. Новые информационно-коммуникационные технологии обучения профессии / А.Н. Волков // Профессиональное образование. – 2001. – № 3. – С.14-16.
3. Воробьев, Г.Г. Твоя информационная культура / Г.Г. Воробьев. – М.: Молодая гвардия, 1988. – 303 с.

4. Кладова, И.С. Универсальность как условие формирования информационно-коммуникационной культуры личности / Кладова И.С. // Актуальные вопросы общественных наук: социология, политология, философия, история: сборник статей по материалам LVI международной научно-практической конференции (16 декабря 2015г.). – Новосибирск, 2015. – С.42-49.

5. Матвеева, Ю.А. Информационно-коммуникативная культура как основная характеристика современного общества / Ю.А. Матвеева // Вестник Бурятского государственного университета. – 2010. – № 6. – С.243-247.

6. Мясоедов, А.М. Медицинская субкультура: специфика, структура, динамика: автореф. дис. ... канд. культур.:24.00.01 / А.М. Мясоедов. – М., 2012. – 29 с.

7. Национальная стратегия устойчивого социально-экономического развития Республики Беларусь на период до 2030 года (НСУР-2030) (одобрена на заседании Президиума Совета Министров Республики Беларусь от 10 февраля 2015 г.) [Электронный ресурс]. – Режим доступа: <http://www.economy.gov.by/ru/macroeconomy/nacionalnaya-strategiya>. – Дата доступа: 20.02.2016.

8. Неумоина, Е.Г. Внедрение информационных технологий в учебный процесс / Е.Г. Неумоина [Электронный ресурс]. – Режим доступа: <http://it-education.ru/2011/section/78/3857/index.html>. – Дата доступа: 21.02.1016.

9. Паршонок, Е.В. Информационно-коммуникационная культура как составляющая социального опыта старшеклассников с нарушениями зрения / Е.В. Паршонок // Специальное образование: опыт и перспективы развития: сб. мат-лов респуб. науч.-практ. конф., г. Могилев, 12 октября 2010 г. – Могилев: УО «МГУ им. А.А. Кулешова», 2011. – С.106-110.

10. Суханов, А.П. Информация и прогресс / А.П. Суханов. – Новосибирск, 1988. – 192 с.

11. Формирование учебной мотивации студентов через повышение информационно-коммуникативной культуры преподавателя / Методические рекомендации [Электронный ресурс]. – Режим доступа: http://ugkr.ru.../recomendacii_po_formirovaniyu_uchebnoy_motivacii. – Дата доступа: 20.02.2011.

E.M. Shevchik-Giris

Polysemy of the Term «Information and Communication Culture»

The article discusses the content of the term «information and communication culture». A definition of the term is given from the viewpoint of cultural studies, pedagogy, sociology, computer science, psychology and medicine. Also it is proposed author's definition of the term.

Статья поступила 30.05.2016



Н.М. Люцко, м. пед. н., библиотекарь I-й категории Научно-педагогической библиотеки учреждения «Главный информационно-аналитический центр Министерства образования Республики Беларусь»

Становление информационно-аналитического обеспечения научной деятельности учреждений высшего образования в условиях университетских библиотек

В статье представлены результаты источниковедческого исследования истории становления элементов информационно-аналитического обеспечения университетской науки средствами библиотек учреждений высшего образования (УВО), раскрыт потенциал для университетских библиотек, а также предложена периодизация развития этой деятельности.

Истоки информационно-аналитического обеспечения научной деятельности ученых приходится на период с 1771 по 1772 гг. и связаны с выпуском указателя рецензируемых книг «Парижская библиография». Это издание стало первым прообразом современных указателей научных ссылок. Одна из первых попыток количественного изучения потока отечественной литературы была предпринята А. Шторхом и Ф. Аделунгом, которые проанализировали собственный указатель литературы за период 1801-1806 гг. по различным параметрам. Еще одним изданием подобного рода является «Указатель ссылок федерального законодательства Шепарда», первый том которого вышел в 1873 г. В начале нового столетия закладываются зачатки методологии и терминосистемы информационно-аналитического обеспечения науки. В 1911 г. русский ученый П. Вальден впервые применил метод анализа цитирования для изучения вклада ученых отдельных стран в развитие химии. Вслед за ним в 1917 г. Ф. Коул и Н. Ильс провели статистический анализ литературы по сравнительной анатомии. Начиная с 1923 г. количественные исследования документопотока стали определять

как «статистическая библиография». Это понятие ввел Е. Хульме, применивший его при ранжировании стран по числу журнальных статей в определенных областях [1, С.77-78].

Оформление информационно-аналитической деятельности сферы науки приходится на 1926 г., когда науковед И.В. Боричевский впервые применил термин «науковедение». Он понимал под таковым «объективные законы развития науки как целостной системы, которые позволяют изучить внутреннюю природу науки и ее роль в обществе» [2, С.5-6]. Это позволило заложить эмпирические основы науковедения. В том же году А. Лотка описал закономерность распределения частот публикаций по авторам в любой области исследований [3], а в 1932 г. Д. Ципф открыл распределение частоты слов естественного языка. Еще через два года С. Бредфорд установил эмпирическую закономерность распределения публикаций по изданиям [4, С.4].

Дальнейшее совершенствование предыдущих разработок в этой области связано с именем Ю. Гарфилда, разработавшего в 1960 г. систему индексирования для научной литературы, основанную на анализе цитирования. Он также изобрел меру количественной оценки научного журнала, именуемую в настоящее время *impact-factor* (IF) и методику его вычисления в базах данных «Science Citation Index» (SCI) [4, С.5]. Что способствовало созданию в 1961 г. Институтом научной информации США первой политематической международной базы данных научного цитирования «Web of Science» (WoS). В 1992 г. WoS была приобретена канадской информационно-издательской корпорацией Thomson Reuters.

Следующим шагом в развитии информационно-аналитического обеспечения научной деятельности стала периодизация науки и расширение понятийного аппарата. В 1963 г. британский историк науки Д. Прайс разделил ее историю на два периода: «малая наука» (отражает разрозненные усилия ученых по наблюдению за окружающим миром и определение закономерностей и постулатов, описывающих функционирование природы и человека) и «большая наука» (характеризуется появлением научных обществ и научных учреждений, которые помогли науке стать управляемым и профессиональным видом деятельности) [5]. А в 1969 г. в науковедческом понятийном аппарате появились сразу два новых термина. А. Причард предложил заменить термин

«статистическая библиография» понятием «библиометрия» [6]. Он понимал под таковой «применение математическо-статистических методов в процессе анализа опубликованной информации». Тогда же В.В. Налимов и З.М. Мульченко впервые ввели в научный оборот понятие «наукометрия», под которой они понимали «совокупность количественных методов изучения науки как информационного процесса» [7]. В том же году Б.С. Брукс разработал уточненную математическую формулировку «закона рассеивания публикаций» С. Бредфорда [1, С.79].

Первыми показателями оценки и развития мировой науки послужили данные о количестве научных статей и их цитируемости, содержащиеся в указателе SCI. Они впервые были отражены в Отчете Национального научного фонда США (ННФ), вышедшем в 1972 г. под названием «Показатели науки». На основе отчетов ННФ и в настоящее время прослеживаются тенденции развития науки во всем мире [8, С.10].

В 1979 г. снова наблюдается тенденция к расширению понятийного аппарата информационно-аналитического обеспечения науки. В статьях немецких ученых Л. Блакерт, С. Шпигель и О. Наке появился термин «информетрия», под которым понималось «использование разнообразного математического аппарата для анализа, выявления закономерностей, формулировки законов информационной деятельности и научной информации, а также для принятия решений в информационной практике» [1, С.77].

Конец 80-х годов XX в. характеризуется созданием национальных индексов научного цитирования на основе IF и SCI. Первый национальный индекс научного цитирования был создан в Китае в 1988 г. [9, С.29]. В дальнейшем собственные индексы цитирования появились в ряде стран Азии, Европы и СНГ.

Со второй половины 90-х гг. XX в. университетские библиотеки начинают заниматься созданием и поддержкой репозиториев научных работ исследователей, а также, объединяясь с издательствами УВО впервые позиционируют себя как издателей научной периодики. Первый репозиторий был создан в 1991 г. библиотекой Корнельского университета США и назывался arXiv.org. А начиная с 1995 г. появляются первые проекты по созданию технологических издательских платформ, таких как: HighWire Press MUSE и т.д. [10, С.6].

Значительным шагом в укреплении научно-издательской деятельности университетских библиотек стало создание в 1997 г. международного библиотечного альянса Scholarly Publishing and Academic Resources Coalition (SPARC), целью которого является консолидированное решение проблем в научно-издательской деятельности [10, С.7-8]. Вовлечение университетских библиотек в эту деятельность потребовало пересмотра структуры библиотечных и издательских баз данных и перехода к онлайн-интегрированному программному обеспечению по созданию, поддержке и предоставлению электронных версий научных изданий в Интернете. Такие платформы были созданы, однако наибольшее распространение в практике университетских библиотек получил разработываемый Public Knowledge Project (PKP) с 1998 г. программный пакет с открытым исходным кодом Open Journal Systems (OJS). Первая версия OJS была представлена в 2001 г. Внедрение в деятельность университетских библиотек OJS и иных компьютерных издательских систем стало предпосылкой к развитию «новой научно-издательской модели». Ученые начали рассматривать библиотеки сквозь призму концепции «library as publisher» (библиотеки как издателя).

Параллельно с научно-издательской деятельностью университетских библиотек в этот период начинает свое развитие еще одна составляющая информационно-аналитического обеспечения – вебометрия. Термин впервые был употреблен в 1997 г. в работе Т. Алминда и П. Ингверсена. Авторы понимали под таковым «раздел информатики, исследующий в соответствии с World Wide Web количественные аспекты создания и использования информационных ресурсов» [11]. Вебометрия послужила толчком к началу исследований в области изучения веб-сайтов университетов мира и, на основе этого, разработки различных рейтингов УВО. Первые исследования подобного были проведены в середине 90-х годов XX века испанской компанией «Cubermetrics Lab». Однако международные рейтинги университетов начали появляться значительно позже. Считается, что вебометрия является составляющей киберметрии, которая занимается изучением природы и характеристик веб-страниц, а также методологию World Wide Web с помощью методов библиометрии. Можно предположить, что понятие «киберметрия» возникло от названия компании Cubermetrics Lab.

Стремительное увеличение числа электронных ресурсов привело к необходимости внедрения унифицированного инструментария для учета различных видов документов (книг, статей, музыкальных файлов и т.д.) в Интернете. Поэтому в 1997 г. возникла необходимость разработки универсального цифрового идентификатора объектов Digital Object Identifier (DOI) – уникальной буквенно-цифровой строки, обеспечивающей возможности постоянной идентификации объекта интеллектуальной собственности [12, С.2]. Первоначально DOI разрабатывалась в качестве инструмента для издательского бизнеса, однако со временем сфера ее применения расширилась. Позднее начали создаваться и иные системы идентификации (OpenID, CrossRef, ResearcherID, Open Researcher & Contributor ID (ORCID) и т.д.).

Несмотря на то, что репозитории научных работ и издательские программные пакеты для обеспечения открытого доступа к информации начали создаваться довольно давно, инициативы и лицензии открытого доступа были разработаны только в начале 2000-х годов XXI в. 2000 г. характеризуется появлением некоммерческого научно-издательского проекта П. Брауна, М. Айзена и Х. Вармуса Public Library of Science (PLoS). PLoS направлена на создание библиотеки журналов и другой научной литературы под свободной лицензией и в открытом доступе. Первый журнал PLOS Biology начал издаваться в октябре 2003 г., все материалы которого доступны по лицензии Creative Commons (CC) Она была создана и выпущена Л. Лессингом, Х. Абельсоном и Э. Элдредом в декабре 2002 г. Тогда же публичным фондом Венгрии «Открытое общество» была принята Будапештская инициатива открытого доступа (Budapest Open Access Initiative) [13], а в 2003 г. на основе нее – Берлинская Декларация об открытом доступе к научному и гуманитарному знанию [14]. В то же время в Институте высшего образования шанхайского университета Цзяо Тун был разработан первый международный рейтинг УВО Academic Ranking of World Universities (ARWU) [15]. После него начали появляться иные популярные в настоящее время рейтинги QS University Rankings, Ranking Web of Universities (Webometrics) и др.

Середина 2000-х гг. характеризуется появлением новых разработок в области отслеживания научного цитирования. В

2004 г. компания Elsevier создала платформу SciVerse Scopus, а корпорация Google – Google Scholar. Эти платформы стали прямыми конкурентами, разработанной в 1961 г. международной базе данных научного цитирования «Web of Science» [16, С.1]. Еще через год американский физик Х. Хирш разработал показатель, ставший количественной характеристикой эффективности деятельности ученого. Показатель основывается на количестве его публикаций и цитируемости этих публикаций [17, С.150]. Его впоследствии стали называть Индексом Хирша (h-index). Тогда же российские ученые В. Глухов и Г. Еременко на базе информационных ресурсов Научной электронной библиотеки eLibrary.ru инициировали формирование Российского индекса научного цитирования (РИНЦ). РИНЦ стал первым национальным индексом среди стран СНГ.

В связи с развитием социальных медиа в 2010 г. Дж. Прием, Д. Тараборелли, П. Грот и С. Нэйлон в качестве альтернативы киберметрии и вебометрии предложили термин «альтметрия» [18], с целью оценивания количества упоминаний, просмотров и скачиваний материалов ученых в социальных сетях, блогах и т.д. Появление альтметрии замкнуло на себе формирование метрических дисциплин, которые включает в себя современное науковедение. В том же году на платформе компании Elsevier SciVerse Scopus появляются новые журнальные метрики – SCImago Journal Rank (SJR) и Source Normalized Impact per Paper (SNIP).

Таким образом, элементы информационно-аналитического обеспечения научной деятельности ученых полностью завершили свое формирование лишь к концу 2010 г. В этот же период ответственность за распространение результатов научной деятельности УВО вызвала интерес ученых-библиотековедов. В 2012 г. Т.А. Колесникова впервые акцентировала внимание на стремлении университетских библиотек к информационно-аналитическому обеспечению научной деятельности УВО. Она подчеркнула, что в этом направлении «скрыт тот потенциал, который позволит библиотекам двигаться вперед в сложных финансовых обстоятельствах» [19, С.4]. В настоящее время множество университетских библиотек стран Запада, Европы и СНГ предпочли продуктивно двигаться в данном направлении работы.

Изучение истории становления информационно-аналитического обеспечения научной деятельности ученых позволило

выявить его условную периодизацию. Формирование началось с конца XVIII в. и прошло три исторических периода: донаушный (1771-1925 гг.) – закладываются зачатки методологии информационно-аналитического обеспечения науки; непосредственно научный (1926 – нач. 90-х гг. XX в.) – формируется терминосистема и основные инструменты; электронно-научный (II пол. 90-х гг. – настоящее время) – развитие информационно-аналитического обеспечения научной деятельности в условиях университетских библиотек с учетом информатизации и открытого доступа к информации.

Литература

1. Редькина, Н.С. Библиометрия: история и современность [Электронный ресурс] / Н.С. Редькина // Молодые в библиотечном деле. – 2003. – № 2. – С.76-86. – Режим доступа: <http://www.twirpx.com/file/778428/>. – Дата доступа: 19.03.2015. – Загл. с экрана.

2. Копанева, Є.О. Наукометричні технології в електронній бібліотеці [Текст]: автореф. дис. ... канд. із соц. комунікацій: 27.00.03 / Є.О. Копанєва; Нац. акад. наук України, Нац. б-ка України ім. В.І. Вернадського. – Київ, 2013. – 15 с.

3. Lotka, A. The frequency distribution on of scientific productivity [Virtual Resource] / A. Lotka. – Access Mode: <https://www.cna.org/sites/default/files/research/5500021600.pdf>. – Date of Access: 19 May 2015. – Title from Screen.

4. Бредихин, С.В. Методы библиометрии и рынок электронной научной периодики [Электронный ресурс] / С.В. Бредихин, А.Ю. Кузнецов. – Новосибирск: ИВМиМГ СО РАН, НЭИ-КОН, 2012. – 256 с. – Режим доступа: http://techlibrary.ru/b1/2i1r1f1e1j1w1j1o_2z.2j.,_2s1u1i1o1f1x1p1c_2h.3m._2u1f1t1p1e2c_1b1j1b1m1j1p1n1f1t1r1j1j_1j_1r2c1o1p1l_2e1m1f1l1t1r1p1o1o1p1k1o1a1u1y1o1p1k_1q1f1r1j1p1e1j1l1j_2012.pdf. – Дата доступа: 19.03.2015. – Загл. с экрана.

5. Прайс, Д. Малая наука, большая наука [Текст] / Д. Прайс // Наука о науке. – 1966. – С.281-384.

6. Prichard, A. Statistical bibliography or bibliometrics? [Virtual Resource] / A. Prichard // J. Doc. – 1969. – Vol. 25. – N 4. – P.348-349. – Access Mode: https://www.academia.edu/598618/Statistical_bibliography_or_bibliometrics. – Date of Access: 19 May 2015. – Title from Screen.

7. Налимов, В.В. Наукометрия. Изучение развития науки как информационного процесса [Электронный ресурс] / В.В. Налимов, З.М. Мульченко. – Режим доступа: <http://www.twirpx.com/file/103528/>. – Дата доступа: 19.05.2015. – Загл. с экрана.

8. Маркусова, В.А. Библиометрия как методологическая и инструментальная основа мониторинга развития и информационной поддержки российской науки [Электронный ресурс]: автореф. дис. ... д-ра пед. наук: 05.25.03 / В.А. Маркусова. – М., 2005. – 48 с. – Режим доступа: <http://tekhnosfera.com/view/53872/a?#?page=12>. – Дата доступа: 19.03.2015. – Загл. с экрана.

9. Копанева, Є. Національні індекси наукового цитування [Електронний ресурс] / Є. Копанева // Бібліотечний вісник. – 2012. – № 4. – С.29-35. – Режим доступа: http://nbuv.gov.ua/j-pdf/bv_2012_4_4.pdf. – Дата доступа: 26.01.2015. – Загл. с экрана.

10. Соловяненко, Д. Академічні бібліотеки у новому соціотехнічному вимірі [Електронний ресурс]: Частина перша. Академічна бібліотека як видавець / Д. Соловяненко // Бібліотечний вісник. – 2010. – № 4. – С.3-14. – Режим доступа: http://nbuv.gov.ua/j-pdf/bv_2010_4_1.pdf. – Дата доступа: 06.10.2012. – Загл. с экрана.

11. Almind, T. Informetric analyses on the World Wide Web [Virtual Resource] : Methodological approaches to «webometrics» / T. Almind, P. Ingwersen // Journal of documentation. – 1997. – Vol. 53. – N 4. – P. 404-426- Access Mode: <http://comminfo.rutgers.edu/~muresan/IR/Docs/Articles/jdocAlmind1997.pdf>. – Date of Access: 19 May 2015. – Title from Screen.

12. Соловяненко, Д. Перспективы внедрения системы DOI в библиотечной сфере [Электронный ресурс] / Д. Соловяненко. – Режим доступа: www.center.crimea.ua/library/solovunenکو.pdf. – Дата доступа: 19.03.2015. – Загл. с экрана.

13. Будапештская инициатива открытого доступа [Электронный ресурс]. – Режим доступа: http://csl.bas-net.by/pdf/24-10-2011/budapesht_decl.pdf. – Дата доступа: 24.04.2015. – Загл. с экрана.

14. Берлинская декларация открытого доступа [Электронный ресурс]. – Режим доступа: http://csl.bas-net.by/pdf/24-10-2011/berlin_declaration.pdf.

10-2011/berlin_decl.pdf. – Дата доступа: 24.04.2015. – Загл. с экрана.

15. About Academic Ranking of World Universities [Virtual Resource] – Access Mode: <http://www.shanghairanking.com/aboutarwu.html>. – Date of Access: 19 May 2015. – Title from Screen.

16. Соловяненко Д.В. Галузь наукометрії в умовах конкуренції основних наукометричних платформ [Електронний ресурс] / Д.В. Соловяненко // Документознавство. Бібліотекознавство. Інформаційна діяльність: проблеми науки, освіти, практики: Зб. матеріалів VIII Міжнар. наук.-практ. конф., Київ, 17-19 трав. 2011 р. – Київ, 2011. – С.180-182. – Режим доступа: <http://archive.nbuv.gov.ua/articles/2011/11sdvgnu.pdf>. – Дата доступа: 06.10.2012. – Загл. с экрана.

17. Глухов, В.А. Информационные ресурсы научной электронной библиотеки elibrary.ru / В.А. Глухов // Электронные ресурсы региона: проблемы создания и взаимоиспользования: матер. регион. науч.-практ. конф. (г. Новосибирск, 25-28 октября 2004 г.). – С.255-258.

18. Altmetrics [Virtual Resource] : a manifesto / J. Priem, D. Taraborelli, P. Groth, C. Neylon. – 2010. – Access Mode: <http://altmetrics.org/manifesto/>. – Date of Access: 19 May 2015. – Title from Screen.

19. Колесникова, Т.А. Новая философия и инновационные направления деятельности библиотеки вуза [Електронний ресурс] / Т.А. Колесникова // Традиції та новації в інформаційному забезпеченні науки та освіти: матеріали міжнар. наук.-практ. конф., Сімферополь, 22-24 трав. 2012 р. – Сімферополь: ТНУ, 2012. – Режим доступа: http://library.zntu.edu.ua/for_librarian/crim_2012/kolesnikova.pdf. – Дата доступа: 26.01.2015. – Загл. с экрана.

N.M. Lutsko

Establishment of the Information and Analytical Support of Scientific Activities of Higher Education Institutions under University Libraries Conditions

The article presents the results of source studies for the history of establishment of elements of information and analytical

support in university research by facilities of libraries of higher education institutions, the potential is disclosed for university libraries and periodization is proposed for the development of this process.

Статья поступила 27.05.2016



Н.А. Балашенко, м.н.с. ГНУ «Институт генетики и цитологии НАН Беларуси», магистрант ГУО «Институт подготовки научных кадров НАН Беларуси»

Информационные технологии в генетике

В статье описывается и группируется по направлениям опыт использования информационных технологий в генетике. В настоящее время в генетике в рамках реализации крупномасштабных проектов получено колоссальное количество данных, требующих обработки. Единственный способ обработки этих данных – привлечение средств информационных технологий.

На сегодняшний день развитие генетических исследований невозможно представить без использования информационных технологий, которые задействованы практически на каждом этапе изучения генетического кода, начиная от поиска литературы, заканчивая анализом огромных объемов информации. Каждый год появляются новые специализированные программные продукты, облегчающие работу ученых, а также предоставляющие в их распоряжение новый инструментарий, что способствует появлению новых видов и технологий исследований. Информационные ресурсы помогают не только рационализировать работу ученых, они также применяются для решения амбициозных задач, которые показались бы научной фантастикой еще двадцать лет назад. Скорость получения данных в генетике на сегодняшний день превышает скорость обработки этих данных. Количество и разнообразие информации, получаемой в результате исследований, делает информационные технологии незаменимыми в генетических исследованиях. Такой подход хорош тем, что компьютеры могут обрабатывать большие объемы данных о последовательности ДНК, генетической экспрессии, аминокислотной последовательности и т.д. Ниже будет рассмотрен ряд наиболее комплексных проблем в области исследований генетических основ клеточной трансформации, которые решаются самыми передовыми средствами информационных технологий.

Как уже было сказано выше, в сфере генетики существует перевес в сторону накопления данных. Это рождает потребность в осуществлении анализа данных большого объема. В англоязычной литературе даже появился термин «the Big Data», под которым обычно подразумеваются способы решения данной проблемы. Основной массив данных, требующий привлечения средств анализа данных большого объема, относится к геномике. Данные, полученные при полногеномном секвенировании, представляют собой набор последовательностей нуклеотидов разной длины – «обрывки генетического кода». Эти последовательности отражают участки генома, но не представляют собой последовательность целиком, как она существует в клетках (так как современные технологии позволяют считывать лишь участки нуклеотидных последовательностей и не позволяют считывать сразу всю последовательность). Полученные последовательности перекрываются, что дает возможность выстроить их в один ряд, однако количество этих перекрывающихся последовательностей таково, что обработка этих данных с помощью программных средств, устанавливаемых на персональный компьютер, весьма затруднительна. Для решения таких задач привлекаются средства облачных технологий, т.е. с помощью конфигурируемых вычислительных ресурсов, доступных через сеть Интернет.

Похожая проблема возникает и при осуществлении РНК-секвенирования, осуществляемого для изучения работы генома. Однако последовательности, которые необходимо восстановить, более короткие. Но после обработки этой информации возникает новая проблема. Работа генов осуществляется путем их взаимодействия между собой и с окружающей средой. В результате образуются сложные генные сети, которые очень сложно анализировать.

Облачные технологии при обработке и анализе последовательностей ДНК

Облачные технологии – это концепция сетевого доступа к устройствам для хранения информации, программным приложениям для ее обработки и иным средствам работы с информацией. В таблице 1 представлены примеры ресурсов, использующих данную концепцию.

Таблица 1

Примеры компаний и учреждений, которые предоставляют возможности для создания, хранения, анализа, визуализации экспериментальных и клинических данных

Компания/ Институт	Предоставляемые возможности	Веб-сайт
Appistry	Доступ к высокопроизводительной большой самоорганизующейся площадке для хранения и обработки клинической информации Appistry	www.appistry.com
BGI	Доступ к BGI – вычислительной платформе, представляющей собой интегрированный сервис, состоящий из универсального программного обеспечения, предназначенного для обработки крупных объемов данных в биоинформатике	www.genomics.cn/en
CLC Bio	CLC Bio имеет платформу, оптимизированную для лучшей производительности. CLC Bio используют собственные алгоритмы, основанные на опубликованных методах, успешно ускоряющих процесс обработки данных	www.clcbio.com
GNS Healthcare	GNS Healthcare разработала масштабный подход к решению проблемы больших объемов данных, которые могли бы быть применены по отрасли здравоохранения	www.gnshealthcare.com
Foundation Medicine	Foundation Medicine находится на переднем крае полногеномных исследований в области онкологии. Является первой платформой для разработки диагностических методов в онкологии, объединения клинические данные и большие аналитические возможности	www.foundationmedicine.com

Knome	Knome анализирует информацию о секвенировании генома с помощью программного обеспечения, позволяющего одновременно изучить и сравнить многие гены, генные сети, и геномы, а также интегрировать другие формы молекулярных и немолекулярных данных	www.knome.com
NextBio	Система передачи данных NextBio позволяет пользователям систематически интегрировать и интерпретировать полученные в других лабораториях и собственные молекулярногенетические данные о модельных организмах и клиническую информацию об отдельных пациентах	www.nextbio.com

Проблема построения и анализа генных сетей в генетических исследованиях и пути ее решения средствами современных информационных технологий

Для изучения взаимодействий генов было разработано множество методических подходов, таких как CRISPR и скрининг shRNA. Анализ данных, полученных в результате использования подходов CRISPR, представляет собой построение генных сетей, отражающих взаимодействия генов в клетке.

Для анализа данных о взаимодействии генов был разработан программный продукт NEST.

Веб-приложения и исходный код NEST находятся в свободном доступе по электронному адресу <http://nest.dfci.harvard.edu>.

Исходный код NEST также дополнительно доступен по адресу <https://github.com/foreverdream2/NEST/releases>.

Моделирование в генетических исследованиях

Основными подходами к моделированию клеточной трансформации являются моделирование динамики клеточной популяции, инициирования и прогрессии опухолей, методы построения филогенетических деревьев для моделирования отношений между клеточными субклонами и вероятностные графические модели для описания зависимостей между мутациями. Для анализа динамики популяций клеток может быть использована эволюционная теория, что позволит сделать вывод об этапах развития, например, опухоли в соответствии с молекулярными данными. Эволюционное моделирование помогает понять, как опухоли возникают, а также спрогнозировать ход развития заболевания и исход медицинских вмешательств.

Метод компьютерной видеомикроскопии живых клеток

Компьютерное микрофотографирование или видеомикроскопия (time-lapse microscopy) – это метод многократного фиксирования изображений микрообъектов, которые получают с помощью микроскопа в сочетании с фото- или видеокамерой, которые в свою очередь передают изображения на компьютер. С помощью специализированного программного обеспечения из набора фотографий создается видеоролик, отражающий процессы, происходящие в клеточной популяции. Компьютерная видеомикроскопия живых соматических клеток позволяет автоматизировать получение и анализ экспериментальных данных о динамике ряда процессов в клеточных популяциях с помощью специального программного обеспечения. Данный метод позволяет наблюдать процессы, происходящие на уровне микрообъектов. Этот экспериментальный подход позволяет изучать динамику различных клеточных событий, что является преимуществом по сравнению с экспериментами с использованием фиксированных цитологических препаратов. Компьютерная видеомикроскопия живых клеток находит все большее применение в разработке клеточных технологий. Выполняемый с помощью данного метода анализ индивидуальных клеток и их клоновых потомств во

времени позволяет изучать пролиферацию и клеточную гибель (апоптоз), а также эпигенетические процессы при онкотрансформации. Компьютерная видеомикроскопия также используется для изучения действия противоопухолевых препаратов на раковые клетки. В отличие от математического моделирования динамики клеточной популяции компьютерная видеомикроскопия позволяет наблюдать процесс напрямую, однако математическое моделирование важно для понимания особенностей динамики роста популяций онкоклеток, а также для прогнозирования в онкологии.

Для более детальной обработки изображений существует специализированное программное обеспечение – как закрытое, так и открытое. Наиболее распространенными пакетами бесплатного программного обеспечения являются «CellProfiler», «ImageJ», «Fiji», позволяющие измерять различные параметры клеток.

Базы данных в генетических исследованиях

Базы данных в рассматриваемой предметной области содержат данные, полученные при исследовании онкологии и онкогенетики. Информация может быть представлена в виде данных, прошедших первичную обработку, также некоторые ресурсы предлагают средства для онлайн-анализа данных и последующей загрузки результатов.

В качестве примера можно привести средство обработки информации MAGI для базы данных TCGA. Этот продукт позволяет осуществлять анализ генетических последовательностей, мутаций. Это веб-приложение с открытым исходным кодом. MAGI позволяет пользователям искать, визуализировать и комментировать большой набор данных, включая данные из проекта «The Cancer Genome Atlas». В дополнение MAGI также позволяет исследователям, загружать данные, полученные самостоятельно и сравнить результаты их обработки с уже имеющимися.

В таблице 2 приведен список баз данных, которые служат специально для исследований генетических основ онкотрансформации. Этот список не содержит баз данных, применяемых во многих предметных областях.

Специализированные базы данных в области
молекулярной генетики рака

Название базы данных	Организация-правообладатель	Вид данных с точки зрения онкогенетики	Объект исследования
The BioExpress® Oncology Suite	Ocimum Bio Solutions, США	Экспрессия генов	Клеточные линии и ткани <i>Homo sapiens</i> , <i>Mus musculus</i> , <i>Rattus norvegicus</i>
Oncomine	Compendia Bioscience, Inc., США	Экспрессия генов	Клеточные линии и ткани <i>Homo sapiens</i>
OncoLand	Omicsoft Corporation, США	Количество копий генов, мутации, метилирование, фосфорилирование, экспрессия генов, микроРНК, белков	Клеточные линии и ткани <i>Homo sapiens</i> , <i>Rattus norvegicus</i> , <i>Mus musculus</i>
ClinicalTrials.gov	National Institutes of Health, США	Разное	Ткани <i>Homo sapiens</i>
Project Data Sphere	The CEO Life Sciences Consortium, США	Разное	Ткани <i>Homo sapiens</i>
Mouse Retrovirus Tagged Cancer Gene Database	Institute of Molecular and Cell Biology, Сингапур	Мутации	Ткани <i>Mus musculus</i>
International Cancer Genome Consortium	–	Мутации	Ткани <i>Homo sapiens</i>
Catalogue Of Somatic Mutations In Cancer (COSMIC)	Wellcome Trust Sanger Institute, Великобритания	Мутации	Ткани <i>Homo sapiens</i>
Network of Cancer Genes	King's College London, Великобритания	Мутации	Ткани <i>Homo sapiens</i>

Oncoreveal	Bopazizi University, Турция	Экспрессия генов	Ткани <i>Homo sapiens</i>
cBio Cancer Genomics Portal	Memorial Sloan-Kettering Cancer Center, США	Количество копий генов, мутации, метилирование, фосфорилирование, экспрессия генов, микроРНК, белков	Ткани <i>Homo sapiens</i>
The Cancer Genome Atlas (TCGA)	National Cancer Institute, США	Количество копий генов, мутации, метилирование, экспрессия генов, микроРНК, белков	Ткани <i>Homo sapiens</i>
Mouse Tumor Biology Database	The Jackson Laboratory, США	Количество копий генов, мутации, экспрессия генов	Ткани <i>Mus musculus</i>
Integrative Oncogenomics Cancer Browser (IntOGen)	Universitat Pompeu Fabra, Испания	Количество копий генов, мутации, экспрессия генов	Ткани <i>Homo sapiens</i>
OncoDB.HCC	Academia Sinica, Тайвань	Количество копий генов, экспрессия генов	Ткани <i>Homo sapiens</i> , <i>Mus musculus</i> , <i>Rattus norvegicus</i>
Progenetix	Universitat Zurich, Швейцария	Количество копий генов	Ткани <i>Homo sapiens</i>
CancerResource	University Medicine Berlin, Германия	Чувствительность к препаратам различных типов опухолей	Ткани <i>Homo sapiens</i>
Roche Cancer Genome Database (RCGDB)	Roche Diagnostics, Penzberg, Германия	Количество копий генов, экспрессия генов	Ткани <i>Homo sapiens</i> , <i>Mus musculus</i> , <i>Rattus norvegicus</i>

Информационно-поисковые системы в сфере генетики

Информационно-поисковые системы в области генетики связаны с базами данных, созданных при выполнении крупномасштабных международных проектов. Наличие этих поисковых систем облегчает работу исследователей и позволяет находить информацию о последовательностях РНК, ДНК, результатах клинических исследований, которую невозможно было бы получить любым другим способом. Большая часть этой информации находится в открытом доступе.

Сайты специализированных журналов

Существует ряд журналов, публикующих исследования в области генетики (таблица 3). Кроме сайтов самих журналов для поиска статей можно использовать онлайн базы данных, такие как <http://www.sciencedirect.com>. Одной из самых удобных систем по поиску статей в области генетики являются базы данных NCBI (The National Center for Biotechnology, электронный адрес – <http://www.ncbi.nlm.nih.gov>). Особенно удобна база данных PMC (<http://www.ncbi.nlm.nih.gov/pmc>), содержащая статьи, находящиеся в открытом доступе.

Таблица 3

Примеры специализированных журналов, публикующих результаты генетических исследований

Название журнала	Сайт журнала
Cancer genetics	http://www.cancerjournal.org/
American Journal of Human Genetics	http://www.cell.com/ajhg/home/
Cell	http://www.cell.com/
BMC Genetics	http://www.biomedcentral.com/bmcgenet/
BMC Medical Genetics	http://www.biomedcentral.com/bmcmedgenet/
Genetics	http://www.genetics.org/
International Journal of Molecular Epidemiology and Genetics	http://www.ijmeg.org/
Molecular Cytogenetics	http://www.molecularcytogenetics.org/
Clinical Epigenetics	http://www.clinicalepigeneticsjournal.com/
Nature	http://www.nature.com/index.html

Заключение

В последние годы произошел невероятный рывок в использовании информационных технологий в сфере генетических исследований. Благодаря быстрому развитию молекулярной генетики изменилось наше понимание процессов, происходящих в клетке. Это приблизило нас к решению многих стоящих перед современной генетикой проблем. В рамках крупномасштабных проектов получено колоссальное количество данных, требующих обработки. Единственный способ обработки этих данных – привлечение средств информационных технологий. Современная генетика немыслима без информационных технологий.

Литература

1. Квитко, О.В. Разработка методов компьютерной видеомикроскопии живых клеток для медицинской трансплантологии, биотехнологии животных и токсикологии / О.В. Квитко, И.И. Конева, Я.И. Шейко, В.Д. Трусова, С.Н. Шевцова, Н.А. Балашенко, А.С. Сапун, С.Е. Дромашко // Молекулярная и прикладная генетика. – 2009. – Т.10. – С.89-100.
2. Николайчик Е.А., Валентович Л.Н. SQ – компьютерная программа для редактирования и анализа биологических последовательностей // Труды Белорусского государственного университета. Физиологические, биохимические и молекулярные основы функционирования биосистем – 2010. – Вып. 5. ч.1. – С.154-162.
3. Beerenwinkel, N. Cancer Evolution: Mathematical Models and Computational Inferenc / N. Beerenwinkel, R. F. Schwarz, M. Gerstung, F. L. Markowetz // Syst. Biol. – 2015. – Vol.64. – P.e1-e25.
4. Costa, F.F. Big Data in Genomics: Challenges and Solutions / F.F. Costa // Laboratory Journal. – 2012. – Vol.11. – P.2-4.
5. Etani, N. Database application model and its service for drug discovery in Model-driven architecture / N. Etani // Journal of Big Data. – 2015. – Vol. 2. – P.1-17.
6. Jiang, P. Network analysis of gene essentiality in functional genomics experiments / P. Jiang, H. Wang, W. Li, C. Zang, B. Li // Genome Biol. – 2015. – Vol. 16. – P.1-10.

7. Herland, M. A review of data mining using big data in health informatics / M. Herland, T. M. Khoshgoftaar, R. Wald // Journal Of Big Data. – 2014. – Vol. 1. – P.1-35.

8. Kunkel, T.A. 2000. DNA replication fidelity / T.A.Kunkel, K.Bebenek // Annu. Rev. Biochem. – Vol. 69. – P.497-529.

9. Toga A.W. Sharing big biomedical data / A.W. Toga, I.D. Dinov // Journal of Big Data. – 2015. – Vol.2 – P.1-12.

10. Weiss, D.G. Videomicroscopy. Light microscopy in biology. A practical approach. / D.G. Weiss, W.Maile, R.A. Wick. – Oxford, 2007. – 32 p.

N.A. Balashenko

Information Technologies in Genetics

The article describes and groups on directions the experience of information technologies in genetics. Currently a huge amount of data received in genetics as part of large-scale projects, and it requires processing. The only way to handle these data is use of information technology.

Статья поступила 27.04.2016



Требования к оформлению статьи для журнала «Информатизация образования»

1. Статья представляется в редакцию на белорусском или русском языке в печатном виде в одном экземпляре. Статья должна быть напечатана на белой бумаге на одной стороне листа со следующими параметрами:

- размеры печатной страницы – А4;
- верхнее, нижнее поля – 2 см;
- левое поле – 3 см;
- правое поле – 1,5 см;
- междустрочный интервал – 1,5;
- гарнитура – Times;
- размер кегля – 14.

Общий объем статьи до 10 страниц.

2. К статье прилагается аннотация (не более 0,5 страницы) для дальнейшего размещения на веб-сайте журнала. Для статей, размещаемых в рубрике «Научные публикации», прилагается также аннотация на английском языке.

3. Вместе с печатным вариантом статьи обязательно предоставляется электронный вариант.

Текст должен быть подготовлен в формате Microsoft Word с параметрами, указанными выше.

Формулы вставляются в текст в формате Microsoft Equation (вставка объекта в Microsoft Word).

Рисунки предоставляются в виде отдельных файлов. При отсутствии файлов возможна вставка рисунков в документ Microsoft Word.

Диаграммы предоставляются в отдельном файле в формате Microsoft Excel с цифровым материалом, по которому были построены диаграммы.

4. Ссылки на литературу даются в квадратных скобках. Перечень источников в порядке появления в тексте приводится под заголовком «Литература» в конце статьи.

5. На отдельном листе необходимо привести сведения об авторах:

- фамилия, имя, отчество (полностью);
- должность и место работы;
- ученое звание;
- номер контактного телефона, адрес электронной почты;
- полный домашний адрес;
- паспортные данные (серия, номер, дата и место выдачи);
- согласие (несогласие) автора на размещение полного варианта статьи на Web-сайте журнала.

6. Статья может быть возвращена автору на доработку или для ее сокращения. В журнале публикуются материалы, получившие положительную рецензию. Рукописи не возвращаются.

Ответственность за то, что материал публикуется впервые, лежит на авторе публикации.

Мнения, высказанные в статьях, отражают точку зрения их авторов и могут не совпадать с мнением редакции.

К публикации принимаются материалы, получившие положительную рецензию.

Рукописи не возвращаются.

Редактор	Е.Н. Кишкурно
Корректор	Е.Н. Кишкурно
Макет и верстка	Д.И. Пунько

Адрес редакции журнала «Информатизация образования»:
220088, г. Минск, ул. Захарова, 59, к. 225.
Тел. 294-15-94. E-mail: elena@unibel.by

Подписано в печать 15.06.2016. Бумага офсетная. Печать цифровая.
Формат 60x84 $\frac{1}{16}$. Усл. печ. листов 6,0.
Тираж 290 экз. Цена свободная. Заказ № 0618.

Отпечатано с готовых диапозитивов
в ОАО «Промпечать».
ЛП № 02330/233 от 11.03.2009.
г. Минск, ул. Черняховского, 3.