

# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ

УДК 372.862

## **Максименкова Ольга Вениаминовна**

*младший научный сотрудник Международной научно-учебной лаборатории интеллектуальных систем и структурного анализа, старший преподаватель департамента программной инженерии факультета компьютерных наук Национального исследовательского университета «Высшая школа экономики», Москва (Россия).*

*E-mail: omaksimenkova@hse.ru*

## **Подбельский Вадим Валериевич**

*доктор технических наук, профессор департамента программной инженерии факультета компьютерных наук Национального исследовательского университета «Высшая школа экономики», Москва (Россия).*

*E-mail: vpodbelskiy@hse.ru*

## **ПРАКТИКА ИСПОЛЬЗОВАНИЯ ОТКРЫТЫХ ДАННЫХ В КУРСЕ «ПРОГРАММИРОВАНИЕ» ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ БАКАЛАВРИАТА «ПРОГРАММНАЯ ИНЖЕНЕРИЯ»**

**Аннотация.** Цель работы – показать возможности использования открытых данных в преподавании курсов программирования.

**Методы и методики.** Результаты внедрения представленной в публикации методики в процесс обучения программированию на первом курсе специальности «Программная инженерия» получены путем сравнительного исследования и проанализированы методами описательной статистики.

**Результаты и научная новизна.** Предложена методика использования открытых наборов данных при разработке тренировочных и контрольных заданий по программированию для студентов бакалавриата информационно-технологического профиля; приведены шаблоны типовых заданий.

Применение открытых данных в учебных целях – достаточно новое направление в образовании, позволяющее значительно улучшить качество подготовки

специалиста в области компьютерных наук: приблизить учебные задачи к реальным; увеличить вариативность контрольных заданий; повысить мотивацию студентов. Однако наряду с позитивными характеристиками метода имеется ряд трудностей его внедрения в учебный процесс. Авторами сформулированы проблемы, возникающие при использовании открытых данных, и показаны возможные пути их решения.

*Практическая значимость.* Полученные в ходе исследования результаты демонстрируют возможности дальнейшего расширения применения открытых данных в образовании. Описанный практический опыт обучения программированию может быть частично или полностью использован преподавателями других учебных курсов.

**Ключевые слова:** открытые данные, открытые образовательные ресурсы, наборы данных, обучение программированию.

DOI: 10.17853/1994-5639-2016-10-107-121

Статья поступила в редакцию 10.08.2016.

Принята в печать 16.11.2016.

#### **Olga V. Maksimenkova**

*Junior Research Fellow, International Laboratory for Intelligent Systems and Structural Analysis; Senior Lecturer, School of Software Engineering, Faculty of Computer Science, National Research University Higher School of Economics, Moscow (Russia).*

*E-mail: omaksimenkova@hse.ru*

#### **Vadim V. Podbelskiy**

*Doctor of Technical Sciences, Professor, School of Software Engineering, Faculty of Computer Science, National Research University Higher School of Economics, Moscow (Russia).*

*E-mail: vpodbelskiy@hse.ru*

## **A PRACTICE OF OPEN DATA ADOPTION TO «PROGRAMMING» COURSE OF «SOFTWARE ENGINEERING» BACHELOR PROGRAM**

**Abstract.** *The aim* of the publication is to show the possibilities of use of open data in teaching courses of programming.

*Methods.* The results of adoption of the technique presented in the publication to the process of training in programming at the first year of the course «Program Engineering» are received by a comparative research and analysed by methods of descriptive statistics.

*Results and scientific novelty.* The technique of the use of open data sets when developing training and control tasks on programming for students of a bachelor degree of information and technological educational program specialization is offered; standard tasks templates are presented.

The application of open data in the educational purposes is a rather new direction in education that considerably enables to improve the quality of training of an expert in the field of computer sciences: to bring closer educational tasks to real; to increase variability of control tasks; to increase motivation of students. However, along with positive characteristics of the presented method, there are some difficulties of the method introduction in educational process. The authors have formulated the problems arising when using open data; possible ways of their decision are shown.

*Practical significance.* The results received during the research show the possibilities of further expansion of application of open data in education. The described practical experience of training in programming can be partially or completely used by teachers of other training courses.

**Keywords:** open data, open educational resources, data sets, teaching programming.

DOI: 10.17853/1994-5639-2016-10-107-121

Received: 10.08.2016.

Accepted for printing: 16.11.2016.

В настоящее время интерес к открытым данным (ОД) высок в различных областях деятельности и на различных уровнях, начиная от специфического профессионального интереса специалистов по анализу информации и заканчивая государственными структурами, в том числе образовательными.

Понимание открытости данных в нашей работе базируется на следующем определении: «Открытыми называют данные, которые могут свободно использоваться, модифицироваться и распространяться любым лицом в любых целях» [1].

Открытые образовательные данные (ООД) (*educational open data*, EOD) рассматриваются исследователями под разными углами. Можно выделить несколько направлений, сгруппированных следующим образом:

1) общие и частные вопросы наук о данных, среди которых накопление, майнинг, обработка и анализ ООД [2, 3];

2) статистические исследования в сфере образования, основанные на ОД [4];

3) использование ОД в преподавании [5, 6]. В этих работах могут применяться не только ООД, но и любые другие данные.

Мы сфокусировались на последнем направлении – возможности применения ОД в преподавании. Областью нашего внимания стало использование ОД для подготовки тренировочных и контрольных заданий по программированию для студентов бакалавриата информационно-технологического профиля.

Идея привлечения ОД для удовлетворения нужд образования сейчас очень популярна. В сообществах, занимающихся вопросами открытого

образования (open education), обсуждаются и исследуются вопросы отождествления некоторых наборов ОД с открытыми образовательными ресурсами (open educational resources, OER) [10]. Существует много работ, описывающих применение ОД при изучении дисциплин, связанных с анализом данных [5, 7–9]. Обычно такие дисциплины преподаются на старших курсах бакалавриата, в магистратуре и на курсах повышения квалификации. Однако нам неизвестны задокументированные примеры использования ОД в дисциплинах, связанных с программированием, при подготовке студентов первого года обучения.

### **Особенности обучения программированию первокурсников информационно-технологических направлений**

Первый год обучения в бакалавриате – один из важнейших этапов подготовки специалиста в области компьютерных наук [12–15]. В сфере образования проблемы студентов-первокурсников составляют отдельный блок исследований. Об этом свидетельствует проведение специализированных конференций и организация специальных секций на научно-педагогических форумах, таких как *Annual Conference of the First-Year Experience*, *International Conference on the First-Year Experience* и подобных.

Указанное обособление не случайно. Так, преподаватели программирования на первом курсе бакалавриата сталкиваются со следующими трудностями:

- естественное разделение всех первокурсников на две группы с существенно разными уровнями подготовки;
- необходимость формирования у студентов тесно связанных теоретических знаний и практических навыков.

Сформированные в процессе обучения навыки должны соответствовать запросам ИТ-индустрии, что в самой простой интерпретации означает: студент должен уметь справляться не только и не столько с типовыми учебными задачами, сколько с реальными вопросами, которые решает ИТ-индустрия.

В связи с этим получили широкое распространение активные образовательные практики, например проблемно-ориентированное (*problem-based learning*) и проектное обучение (*project learning*). Они предполагают использование учебно-тренировочных материалов, тесно связанных с областями практической и научной деятельности, что подразумевает вовлечение студентов в реальные индустриальные проекты и моделирование учебных ситуаций, максимально приближенных к реальности.

Мировая тенденция к укреплению связей между образованием и индустрией вылилась в активно набирающую популярность инициативу «Задумать – Спроектировать – Реализовать – Управлять» (*Conceive – De-*

*sign – Implement – Operate* (CDIO)), позиционируемую как новый подход к проектному обучению в инженерном образовании.

### Работа с данными в курсах программирования

Обязательной частью почти каждого курса программирования является подготовка студентов к работе с различными источниками данных. В контексте первого курса это может подразумевать работу с файлами, на старших курсах – работу с базами данных (*data base*), хранилищами данных (*data storage*), веб-службами (*web-service*) и т. д.

Современные программные системы и комплексы оперируют данными, представленными разнообразными форматами. Таким образом, современный ИТ-специалист должен быть знаком не только с простым текстовым форматом файлов, но также с форматом табличного представления *comma-separated-values* (CSV) и распространенными сериализованными форматами, такими как XML, JSON и др. (см. краткий обзор в табл. 1).

Таблица 1

Некоторые форматы представления данных

Формат	Ссылка
1	2
<b>CVS</b>	CSV – Comma Separated Values [ <a href="http://data.okfn.org/doc/csv">http://data.okfn.org/doc/csv</a> ] RFC 4180 – Common Format and MIME Type for Comma-Separated Values (CSV) Files [ <a href="http://tools.ietf.org/html/rfc4180">http://tools.ietf.org/html/rfc4180</a> ] RFC 7111 – URI Fragment Identifiers for the text/csv Media Type [ <a href="https://tools.ietf.org/html/rfc7111">https://tools.ietf.org/html/rfc7111</a> ]
<i>Пример содержимого cvs-файла</i>	
ГП № 107 филиал № 2 «ДЗМ»; «Северо-Восточный административный округ»; «район Отрадное»; «127566»;	
<b>XML</b>	XML 1.0 – W3C Recommendation 2008–11–26 [ <a href="http://www.w3.org/TR/REC-xml">http://www.w3.org/TR/REC-xml</a> ] Namespaces in XML 1.0 (Third Edition) W3C Recommendation 2009–12–8 [ <a href="http://www.w3.org/TR/REC-xml-names">http://www.w3.org/TR/REC-xml-names</a> ] Document Object Model (DOM) Level 2 Core Specification (Version 1.0) W3C Recommendation 2000–11–13 [ <a href="http://www.w3.org/TR/DOM-Level-2-Core">http://www.w3.org/TR/DOM-Level-2-Core</a> ]
<i>Пример содержимого xml-файла</i>	
<pre>&lt;? xml version="1.0"?&gt; &lt;University xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"&gt; &lt;UniversityName&gt;NRU HSE&lt;/UniversityName&gt; &lt;Departments&gt; &lt;DeptItem DeptName="SE"&gt; &lt;Staff&gt; &lt;StaffItem xsi:type="Professor" HumanName="Ivanov" /&gt; &lt;StaffItem xsi:type="Professor" HumanName="Petrov" /&gt;</pre>	

1	2
	<pre> &lt;/Staff&gt; &lt;/DeptItem&gt; ... &lt;/Departments&gt; &lt;/University&gt; </pre>
<b>JSON</b>	JSON [http://json.org] JSON-ECMA-404 [http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf]
<i>Пример содержимого json-файла</i>	
	<pre> {"Departments": [ {"DepartmentName": "SE", "StaffList": [ {"_type": "Professors:#JSONSerialization", "Name": "Ivanov"}, {"_type": "Professors:#JSONSerialization", "Name": "Petrov"} ]} ], "UniversityName": "NRU HSE"} </pre>

Перечисленные в табл. 1 форматы типичны для наборов ОД. Поскольку информация, представленная в CSV, XML или JSON форматах по существу является текстовой, т. е. может быть получена стандартными средствами работы с текстовыми файлами, использование ОД является одним из простых и доступных способов модернизации учебного процесса и приближения учебных задач к подзадачам современной ИТ-индустрии.

В качестве типового задания по дисциплине «Программирование» могут быть полностью или частично использованы задачи, относящиеся к области наук о данных, например: работа с «сырыми» данными (*raw data*), включая недавно появившееся направление – очистку данных (*data cleaning*) [16, 17]; задачи подготовки данных (*data preprocessing*) и др.

Рассмотрим опыт применения ОД при разработке и использовании типовых и контрольных задач по дисциплине «Программирование» для первого курса бакалаврской программы «Программная инженерия» факультета компьютерных наук Национального исследовательского университета «Высшая школа экономики» (НИУ ВШЭ).

### **Методика внедрения заданий в учебный процесс**

Впервые задания с применением наборов ОД были введены в практику контроля на факультете компьютерных наук НИУ ВШЭ в третьем модуле 2014/15 уч. г. и использованы в качестве основы для контрольного домашнего задания (КДЗ), для подготовки к выполнению которого в материалы практических занятий по программированию был добавлен новый материал по работе с csv-файлами. Выдача и сбор заданий были организованы посредством системы поддержки учебного процесса (*Learning Management System*, ЛМС) университета. Подробно процесс подготовки, проведения и проверки КДЗ представлен на рис. 1.

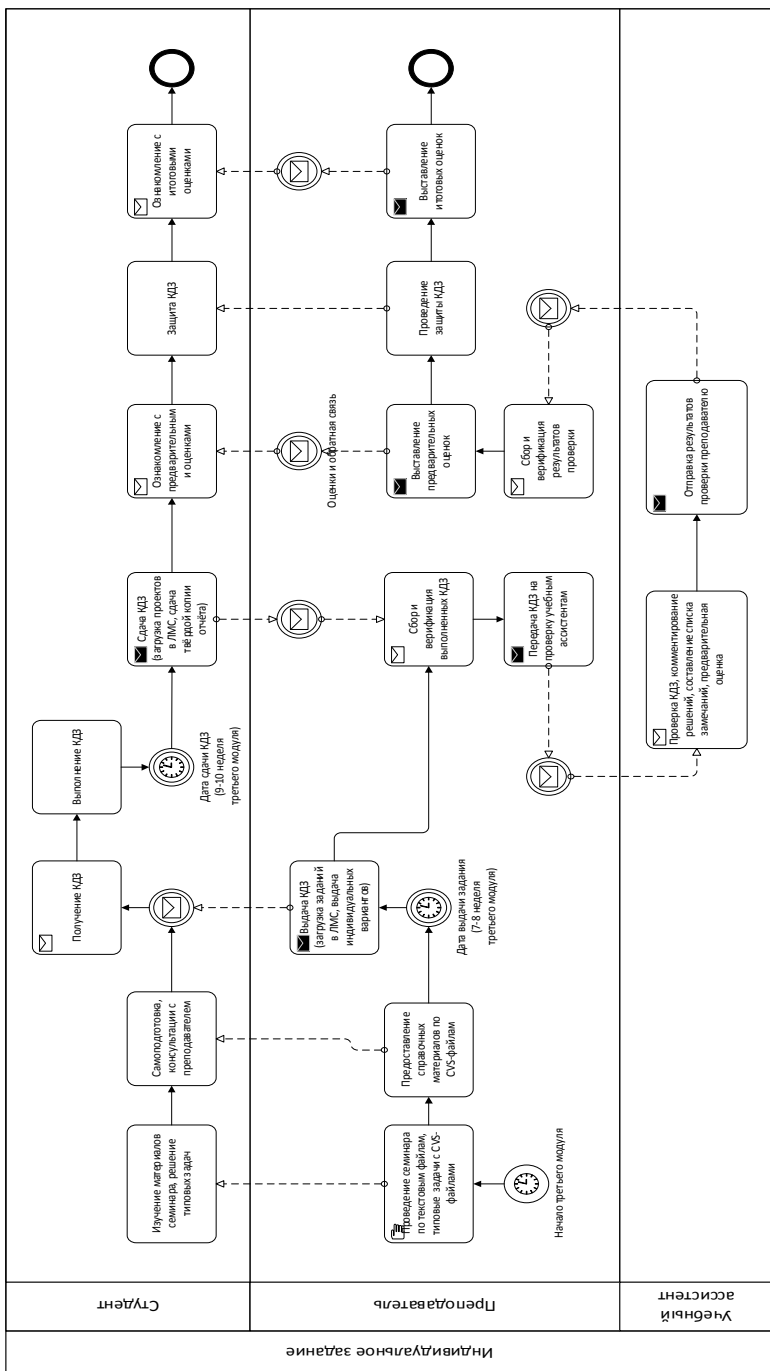


Рис. 1. Последовательность использования заданий в курсе «Программирование»

Для создания заданий были выбраны данные, представленные на портале «Открытые данные правительства Москвы»<sup>1</sup>. Всего было отобрано 15 наборов, эквивалентных по объему и структуре хранящихся данных. Под эквивалентностью по структуре подразумевается наличие в файле информации, которая в программе может быть представлена различными типами данных, например строками и числами.

Типовое КДЗ предполагает разработку на языке С# библиотеки классов с методами, функционально эквивалентными встроенным командам обработки структур `data.frame` и `data.table` языка программирования R (таких как `head` – получить первые записи из набора, `tail` – получить последние записи из набора) [18]. Пример типового КДЗ дан в табл. 2.

Таблица 2

Пример формулировки типового КДЗ по теме «Текстовые файлы»

<b>Вспомогательная информация по чтению из текстовых файлов</b>
1. How to read from and write to a text file by using Visual C# [ <a href="http://support.microsoft.com/kb/816149">http://support.microsoft.com/kb/816149</a> ] 2. <code>StreamReader.ReadLine</code> – метод [ <a href="https://msdn.microsoft.com/ru-ru/library/system.io.streamreader.readline(v=vs.110).aspx">https://msdn.microsoft.com/ru-ru/library/system.io.streamreader.readline(v=vs.110).aspx</a> ]
<b>Написать библиотеку классов, включающую методы для обработки csv-файла <code>SquareCamera.csv</code> (имя файла передается в качестве параметра):</b> 1) Метод <code>ReadAndPrint()</code> читает и выводит в консольное окно все строки файла. 2) Метод <code>HeadArray()</code> возвращает массив строк, содержащий первые N строк файла, за исключением строки заголовка, N – параметр метода. 3) Метод <code>TailArray()</code> возвращает массив строк, содержащий последние N строк файла, N – параметр метода. 4) Метод <code>GetDistrictArray()</code> , возвращающий массив, элементы которого – строки с названиями районов, в которых размещены камеры наблюдения. 5) Метод <code>DistrictIntInfo()</code> , возвращающий массив строк, содержащих информацию о названии района и общем количестве камер наблюдения в нем
Файл <b><code>SquareCamera.csv</code></b> получен с сайта: <a href="http://data.mos.ru/datasets/1498">http://data.mos.ru/datasets/1498</a>

КДЗ выдается студентам после семинаров по работе с текстовыми файлами, по отношениям между классами и по решению типовых задач на обработку csv-файлов. Содержание КДЗ предполагает разработку программы, позволяющей получать данные из csv-файла, создавать объекты классов с использованием этих данных, записывать результаты обработки в csv-

<sup>1</sup> Портал открытых данных правительства Москвы: <http://data.mos.ru>.



файлы. Комплект методических материалов для КДЗ включает комплект индивидуальных заданий, «Требования к контрольному домашнему заданию», размещенные в ЛМС, и cvs-файл с индивидуальным набором ОД (табл. 3).

Таблица 3

Пример индивидуального варианта к контрольному домашнему заданию по теме «Текстовые файлы. Отношения между классами»

#### **Вариант 10**

Программа Контрольного домашнего задания представляет собой WindowsForms приложение и предназначена для просмотра, обработки и сохранения результатов обработки данных из файла:

«Поликлиническая помощь взрослым: версия 2.4 от 20.10.2014.csv»

##### *1. Требования к основным классам приложения*

1.1. Основная информация о взрослых поликлиниках хранится в объектах класса «Поликлиника». Набор полей класса задается полями csv-файла, кроме полей, содержащих адрес и телефон. Адрес представлен полем типа «Контакты».

1.2. Класс «Контакты» представляет адрес (индекс, город, улица, номер дома) и телефон.

1.3. Класс «Поликлиника» находится в отношении композиции с классом «Контакты»

1.4. Дополнительные классы, необходимые для решения задачи, определяет исполнитель задания.

##### *2. Требования к функциям приложения*

2.1. Программа должна позволять:

2.1.1. Открыть \*.csv файл с исходными данными. Использовать элемент управления OpenFileDialog.

2.1.1.1. *Ограничение.* В случае ошибок открытия файла или обнаружения некорректных данных, программа должна выводить сообщение.

2.1.1.2. Отсортировать данные по полям: District, ShortName

2.1.3. Отфильтровать данные по полям исходного csv файла: PaidServiceInfo, AdmArea

2.1.4. Сохранять результаты сортировок и фильтров в файле формата csv. Использовать SaveFileDialog.

2.1.4.1. *Ограничение.* В случае ошибок открытия файла или появления некорректных данных, программа должна выводить сообщение.

2.1.4.2. *Режимы сохранения в файл*

2.1.4.2.1. Создание нового файла.

2.1.4.2.2. Замена содержимого уже существующего файла.

2.1.4.2.3. Добавление данных к содержимому уже существующего файла.

##### *3. Требования к интерфейсу*

3.1. Использовать элемент управления «Сетка данных» – DataGridView

3.2. Количество отображаемых в сетке элементов ( $N$ ) выбирается пользователем,  $N > 1$  и не должно превышать количества записей в csv файле.

##### *4. Требования к устойчивости приложения*

4.1. Аварийные ситуации, возникающие при выполнении приложения, должны обрабатываться, пользователю должны выводиться информативные сообщения.

Необходимость работы с реальными данными требует от студентов во время выполнения КДЗ проводить адаптацию программного кода согласно особенностям полученных ими наборов. Обратная связь от студентов в этот период предоставлялась очно на семинарах или отправлялась по электронной почте. Ниже приведены примеры вопросов, задаваемых студентами:

**Студент 1:** «Вдогонку еще один вопрос: в csv-данных первый столбец ROWNUM, содержащий номера строк, не несет уникальной информации о строках. Мы можем считать данные в List <наши объекты> и воспринимать ROWNUM при выводе как индекс соответствующего объекта в List'e (увеличенного на 1). Таким образом, эти индексы будут перестраиваться автоматически, например, при удалении объекта из списка.

*Вопрос: это правда?»*

**Студент 2:** «В интернете я нашел следующее обсуждение, где пытались понять, какой из двух способов чтения данных из файла лучше:

*<http://stackoverflow.com/questions/8037070/whats-the-fastest-way-to-read-a-text-file-line-by-line>.*

*Мнения, к сожалению, разделились. Скажите, что из этих двух способов Вы можете порекомендовать к использованию в КДЗ?»*

Если обобщить вопросы, возникающие у учащихся, можно выделить следующие проблемы, с которыми сталкиваются студенты:

- 1) особенности оформления csv-файлов (например, применение различных разделителей для данных; присутствие разделителей в экранированных двойными кавычками строках и пр.);
- 2) структура и состав данных (например, пропущенные данные, нарушенный формат представления и т. п.);
- 3) представление данных в программных объектах.

## **Параметры апробации и методы анализа результатов**

При апробации разработанной нами методики 99 студентов первого курса бакалавриата, обучавшихся в 2015 г. на факультете компьютерных наук НИУ ВШЭ, были разделены на две группы:

- 1) экспериментальную (57 человек), получившую задания, требующие обработки реальных наборов ОД;
- 2) контрольную (42 человека), где были предложены задания с текстовыми файлами, содержащими данные в искусственном формате.

Оценки за КДЗ выставлялись согласно принятой в НИУ ВШЭ шкале: 0 – работа отсутствует, 1–3 – неудовлетворительно, 4–5 – удовлетворительно, 6–7 – хорошо, 8–10 – отлично. Эта шкала относится к порядковым (*ordinal scale*) [19], следовательно, при обработке результатов были доступны для вычисления только средние и процентильные ранги.

Основной целью нашей статьи является описание методики применения наборов ОД в программистских курсах, поэтому при анализе результатов ограничимся описательной статистикой количественных и анализом качественных данных, полученных в результате обратной связи по контрольному мероприятию от преподавателя к студенту.

## Результаты

Анализ содержания обратной связи по КДЗ, предоставленной 57 студентам экспериментальной группы, показал, что все они (100%) успешно справились

- с проектированием классов и организацией отношения между ними;
- открытием csv-файла и чтением его записей;
- представлением данных с использованием предложенного в задании элемента управления.

Наиболее распространенными проблемами стали ошибки при создании в программе csv-файла и открытии cvs-файла, созданного при помощи программы. Такие ошибки допустили 25% студентов. У 20% учащихся при проектировании классов был нарушен принцип инкапсуляции (не в рамках организации наследования).

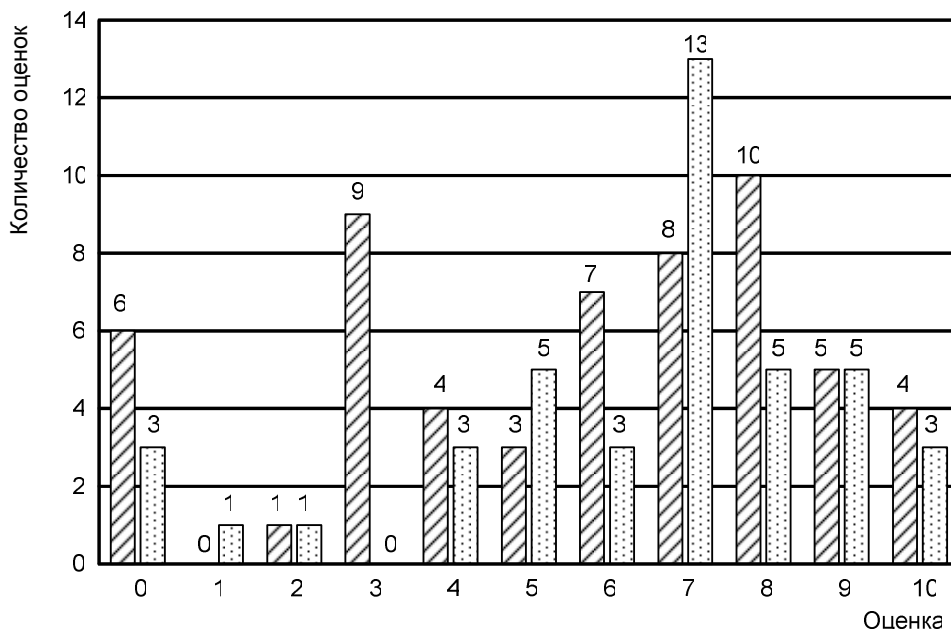


Рис. 2. Распределение оценок за выполнение КДЗ в экспериментальной и контрольной группах:  
▨ – экспериментальная группа; ▤ – контрольная группа

Обобщенные данные о результатах выполнения КДЗ в экспериментальной и контрольной группах (рис. 2) показывают, что явного различия по количеству неудовлетворительных оценок нет. Средние значения по оценкам экспериментальной группы (5,67) и контрольной группы (6,29) находятся немного выше среднего значения шкалы (5). Распределение для сгруппированных по четырехбалльной шкале оценок показано в табл. 4 (несданные работы не учитывались).

Таблица 4

Распределение сгруппированных оценок в группах

Группа	Оценка			
	неудовлетворительно	удовлетворительно	хорошо	отлично
Экспериментальная	10	7	15	19
Контрольная	2	8	16	13

Доля удовлетворительных и хороших оценок преобладает как в экспериментальной (43%), так и в контрольной (62%) группе. Пропорции отличных оценок в них (37% и 33% соответственно) практически эквивалентны. Наблюдается ощутимый выброс неудовлетворительных оценок в экспериментальной (20%) группе по сравнению с контрольной (5%). Наличие выброса может быть обусловлено несколькими причинами. Во-первых, группы не были рандомизированы и выборки не были усреднены по текущей успеваемости. Во-вторых, при планировании эксперимента отсутствовала калибровка преподавателей (как экспертов) экспериментальной и контрольной группы. В-третьих, в критериях при оценивании работ на границе баллов 3 и 4 может присутствовать неоднозначность.

Анализ результатов контрольного домашнего задания третьего модуля 2014/15 уч. г. наглядно показал необходимость пересмотра и доработки некоторых критериев оценивания<sup>1</sup>, особенно на границах неудовлетворительных и удовлетворительных оценок, а также поиска методики калибровки преподавателей учебной дисциплины.

Обобщая все вышеизложенное, отметим, что полученные результаты демонстрируют возможности дальнейшего расширения применения ОД в образовании. Например, мы предполагаем проведение сравнительного исследования результатов с привлечением данных прошлых лет, чтобы удостовериться в том, что результаты обучения в целом не ухудшились.

<sup>1</sup> Критерии оценивания работ, содержащих программную реализацию, представлены в программе учебной дисциплины «Программирование», актуальная версия которой доступна на странице: <https://www.hse.ru/edu/courses/152253017>.

Опыт использования ОД в учебных целях является достаточно новым направлением. Этот метод позволяет значительно улучшить качество подготовки специалиста в области компьютерных наук:

- приблизить типовые задачи к реальным;
- увеличить вариативность контрольных заданий;
- повысить мотивацию студентов.

Однако наряду с позитивными характеристиками метода следует обозначить ряд трудностей его внедрения в учебный процесс:

- необходимость подбора эквивалентных по сложности наборов данных;
- потребность в отслеживании версий наборов данных;
- низкое качество отдельных наборов данных.

Описанный в данной статье практический опыт применения открытых данных при обучении программированию может быть частично или полностью использован преподавателями других учебных курсов.

*Статья рекомендована к публикации  
д-ром техн. наук, проф. В. А. Копновым*

## **Литература**

1. Open Knowledge project. The open definition [Электрон. ресурс]. Режим доступа: <http://opendefinition.org> (дата обращения: 8.01.2015).
2. Encheva S. Lecture notes in electrical engineering // Educational data mining for problem identification. 2014. Vol. 269. P. 3491–3494.
3. Muna A. R., Atheer S. A. K., Hend S. A. K. Lecture notes in electrical engineering // Educational data mining: a systematic review of hte published literature 2006–2013. 2015. Vol. 285. P. 711–720.
4. EdStats: Education Statistics [Электрон. ресурс] Режим доступа: <http://datatopics.worldbank.org/education/> (дата обращения: 10.02.2015).
5. Drupal Groups. Using open data in education [Электрон. ресурс]. 2011. Режим доступа: <https://groups.drupal.org/using-open-data-education> (дата обращения: 25.01.2015).
6. Ernst M. Teaching intro CS and programming by way of scientific data analysis [Электрон. ресурс] // PATRAT: program analysis, the practice and theory. Michael Ernst's technical blog: Режим доступа: <http://programanalysis.blogspot.ru/2012/08/teaching-intro-cs-and-programming-by.html> (дата обращения: 3.03.2015).
7. Jormanainen I., Sutinen E. Koli Calling'13 // An open approach for learning educational data mining. Koli. 2013. P. 203–204.
8. Jo S., Ku J. O. Problem based learning using real-time data in science education for the gifted // Gifted education international. 2011. Vol. 27. P. 263–273.
9. White S. Conceptual Structures for STEM data: Linked, Open, Rich and Personal // ICCS 2013. 2013. P. 1–21.
10. Radchenko I., Sakoyan A. On Some Russian Educational Projects in Open Data and Data Journalism // Open Data for Education. Linked, Shared, and Reusable Data for Teaching and Learning. Springer International Publishing, 2016. P. 153–165.

11. Maksimenkova O. V., Podbelskiy V. V. 10th International Conference on Computer Science & Education // On practice of using open data in construction of training and assessment tasks for programming courses. 2015. P. 233–236.
12. Jackson D., Miller R. A new approach to teaching programming. 2009.
13. Vahrenhold J., Paul W. Developing and validating test items for first-year computer science courses // Computer science education. October 2014. Vol. 24. № 4. P. 304–333.
14. Pendergast M. O. Teaching Introductory Programming to IS Students: Java Problems and Pitfalls // Journal of Information Technology Education. 2006. Vol. 5. P. 491–515.
15. Leutenegger S., Edgington J. A. 38th SIGCSE Technical Symposium on Computer Science Education, SIGCSE 2007 // A games first approach to teaching introductory programming. New York. 2007. P. 115–118.
16. De Jonge E., van der Loo M. An introduction to data cleaning with R. Heerlen: Statistics Netherlands, 2013. 53 p.
17. Wichham H. Tidy Data // Journal of Statistical Software. 2014. Vol. 59. № 10.
18. What is R? [Электрон. ресурс] // The R Project for Statistical Computing. Режим доступа: <https://www.r-project.org/about.html> (дата обращения: 09.12.2014).
19. Stevens S. S. On the Theory of Scales of Measurement // Science. 1946. Vol. 103. № 2684. P. 677–680.

## References

1. Open Knowledge project. The open definition. Available at: <http://open-definition.org>. (Translated from English)
2. Encheva S. Lecture notes in electrical engineering. *Educational Data Mining for Problem Identification*. 2014. Vol. 269. P. 3491–3494. (Translated from English)
3. Muna A. R., Atheer S. A. K., Hend S. A. K. Educational data mining: a systematic review of the published literature 2006–2013. *Lecture Notes in Electrical Engineering*. 2015. Vol. 285. P. 711–720. (Translated from English)
4. EdStats: Education Statistics (2015). Available at: <http://datatopics.worldbank.org/education/>. (Translated from English)
5. Drupal Groups. Using open data in education. 2011. Available at: <https://groups.drupal.org/using-open-data-education>. (Translated from English)
6. Ernst M. Teaching intro CS and programming by way of scientific data analysis. 2012. Available at: <http://programanalysis.blogspot.ru/2012/08/teaching-intro-cs-and-programming-by.html>. (Translated from English)
7. Jormanainen I., Sutinen E. An open approach for learning educational data mining. *Koli Calling'13*. Koli. 2013. P. 203–204. (Translated from English)
8. Jo S., Ku J. O. Problem based learning using real-time data in science education for the gifted. *Gifted Education International*. 2011. Vol. 27. P. 263–273. (Translated from English)
9. White S. Conceptual Structures for STEM data: Linked, Open, Rich and Personal. *ICCS*. 2013. P. 1–21. (Translated from English)
10. Radchenko I., Sakoyan A. On Some Russian Educational Projects in Open Data and Data Journalism. *Open Data for Education. Linked, Shared, and*

*Reusable Data for Teaching and Learning*. Springer International Publishing, 2016. P. 153–165. (Translated from English)

11. Maksimenkova O., Podbelskiy V. On practice of using open data in construction of training and assessment tasks for programming courses. *10th International Conference on Computer Science & Education*. 2015. P. 233–236. (Translated from English)

12. Jackson D., Miller R. A new approach to teaching programming, 2009. (Translated from English)

13. Vahrenhold J., Paul W. Developing and validating test items for first-year computer science courses. *Computer Science Education*. 2014, October. Vol. 24. № 4. P. 304–333. (Translated from English)

14. Pendergast M. O. Teaching Introductory Programming to IS Students: Java Problems and Pitfalls. *Journal of Information Technology Education*. 2006. Vol. 5. P. 491–515. (Translated from English)

15. Leutenegger S., Edgington J. A. A games first approach to teaching introductory programming. *38th SIGCSE Technical Symposium on Computer Science Education*. SIGCSE 2007. New York, 2007. P. 115–118. (Translated from English)

16. de Jonge E., van der Loo M. An introduction to data cleaning with R. Heerlen: Statistics Netherlands, 2013. 53 p.

17. Wichham H. Tidy Data. *Journal of Statistical Software*. 2014. Vol. 59. № 10. (Translated from English)

18. What is R? The R Project for Statistical Computing. 2014. Available at: <https://www.r-project.org/about.html>. (Translated from English)

19. Stevens S. S. On the Theory of Scales of Measurement. *Science*. 1946. Vol. 103. № 2684. P. 677–680. (Translated from English)

УДК 009.1082

### **Botagoz T. Kerimbaeva**

*Associate Professor, Khoja Ahmet Yasawi International Kazakh-Turkish University, Republic of Kazakhstan, South Kazakhstan Region, Turkistan (Kazakhstan).*

*E-mail: Kerimbaeva-bota@mail.ru*

### **Karligash Kaya**

*Master of Degree, Teacher, Bahcesehir Science and Technology High School, Diyarbakir (Turkey).*

*E-mail: karligas.kaya@bahcesehir.k12.tr*

## **TO THE QUESTION OF THE USING OF INFORMATION-COMPUTER TECHNOLOGIES IN LEARNING ENGLISH LANGUAGE**

**Abstract.** *The aim of the article is the using of informational-computer technologies in learning English language of future specialists very effectively, as the didactic function of these technologies is wide. This is due to the fact that*