



М.С. Долинский, к.т.н., доцент кафедры математических проблем управления Гомельского государственного университета им. Ф.Скорины,
Ю.В. Решетько, магистрант Гомельского государственного университета им. Ф.Скорины,
М.А. Долинская, магистрант Гомельского государственного университета им. Ф.Скорины

Флеш-шаблоны для создания заданий развивающего обучения

Система дифференцированного развивающего обучения программированию [1-8], базирующаяся на комплексе Интернет-курсов «Учимся думать», «Начинаем программировать», «Базовое программирование» в рамках проекта «Дистанционное обучение в Беларуси» (<http://dl.gsu.by>, далее DL), включает в себя тысячи заданий, различных как по содержанию, так и по способам их выполнения. В данной статье описываются флеш-шаблоны, позволяющие, с одной стороны, в десятки раз сокращать трудозатраты на создание обучающих заданий, а с другой стороны, делать эти задания яркими, интересными и привлекательными даже для младших школьников. Флеш-шаблон – это программа, выполненная на базе технологии Macromedia Flash и обеспечивающая определенную функциональность путем считывания описания задания из текстового файла. При этом форматы этих файлов предельно просты, поэтому создание нового задания может

занимать всего несколько минут не только у учителя, но даже у младшего школьника. Сами флеш-шаблоны, их описание и примеры применения находятся в открытом доступе на сайте DL в специальном курсе «Создание Flash-заданий для DL». Таким образом, любой желающий может, воспользовавшись ими, создавать и использовать новые задания, как на локальном компьютере, так и загружая их в систему DL.

Раскраска квадратная

Задание заключается в том, что нужно раскрасить справа картинку по образцу, представленному слева (рис. 1). Для формирования задания (картинки слева) необходимо указать в специальном текстовом файле цвет каждой клеточки, обозначив его числом, соответствующим цвету. Для задания клеточек из двух треугольников числа задаются через одну из соответствующих наклонных, например 6/7 или 7\6. Кроме того, специальным параметром можно задавать реакцию на

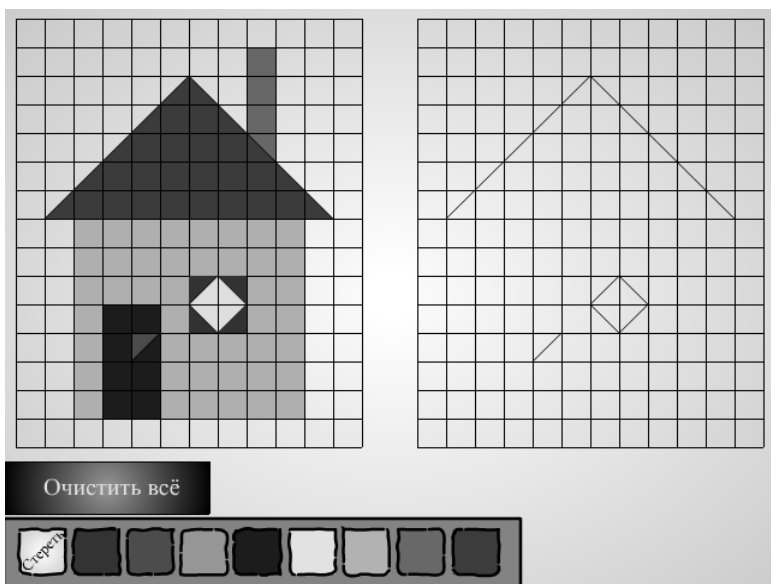


Рис. 1. Задание «Раскрась домик»

ошибку: не закрашивать и подавать звуковой сигнал об ошибке, медленно стирать ошибочный цвет, не сообщать об ошибке закраски конкретной клеточки – но не принимать задание, пока есть хотя бы одна ошибочно закрашенная клеточка. При выполнении задания имеется возможность стирать цвет в некоторой (ошибочно закрашенной) клетке или очистить все закрашенные клетки.

Раскраска круглая

Аналогично предыдущему, данное задание заключается в раскраске правого рисунка по образцу, представленному в левом рисунке (рис. 2). Отличие заключается в том, что ячейки имеют круглую форму. Аналогичные задания продаются в реальном исполнении, когда маленькие цветные трубочки нужно одевать на стерженьки. Они то и послужили прообразом компьютерного варианта задания. Как и в квадратной раскраске, цвет ячейки задается соответствующим числом, имеется параметр реакции на ошибку ученика (не рисовать, стирать, не давать подсказки об ошибке закраски конкретной ячейки), поддерживается возможность стереть цвет в конкретной ячейке или во всех закрашенных ячейках. Кроме того, размер ячейки также является одним из параметров задания.

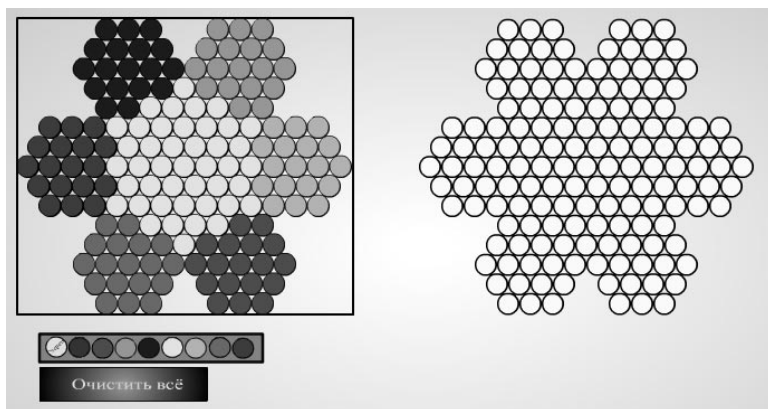


Рис. 2. Задание «Раскрась цветочки»

Отличия

В этом задании необходимо сравнить две «почти» одинаковые картинку и кликами мыши указать отличия (рис. 3). В случае правильного клика на обоих рисунках появляется отметка о найденном отличии, кроме того, слева внизу появляется галочка. В случае ошибки – справа внизу появляется крестик. Количество допустимых ошибок является параметром задания. При превышении допустимого числа ошибок, рабочее поле очищается от всех отметок и задание необходимо выполнять сначала. Если ученик затрудняется в поиске отличий, он может кликнуть на поле «Не знаю» и на обоих рисунках появится квадратная сетка, автоматически выделяются рамкой соответствующие квадратики обоих рисунков. Клавишами «Стрелки» вправо, вверх, вниз, влево их можно перемещать синхронно по обоим картинкам. Тем самым, сравнивать можно не картинку целиком, а только выделенные их части. Если такой подсказки оказалось недостаточно, и ученику помогли найти отличия, учитель может попросить его нажать кнопку «Сначала» для того, чтобы ученик выполнил эту работу еще раз самостоятельно. Автором задания задаются количество отличий, координаты и радиус (в пикселях) областей отличий, количество строк и столбцов разбиения рисунка на фрагменты.



Рис. 3. Задание «Найди отличия»

Составление раскраски

На рисунке 4 слева представлен внешний вид соответствующего задания, справа – результат его правильного выполнения. То есть необходимо составить заданную область фигурками разного цвета. При этом с помощью мыши их можно перемещать, а с помощью клавиш «Стрелки» поворачивать на 90 градусов. Ниже приведен полный текст файла задания:

```
&Stroka=10&
&Mass1=_ _ _ _ a a _ _ _ _.&
&Mass2=_ _ _ f f a f _ _ _.&
&Mass3=_ _ a a f a f f _ _.&
&Mass4=_ c a a b b f a a _.&
&Mass5=c c a a f f f a a b.&
&Mass6=c c a a f f f a a b.&
&Mass7=_ c a a a a a a _.&
&Mass8=_ _ c b a b b a _ _.&
&Mass9=_ _ _ b a a a _ _ _.&
&Mass10=_ _ _ _ a a _ _ _ _.&
```

Первая строка этого файла задает количество строк в задании. Далее символ «_» задает незакрашенный квадрат, символы «a», «b», «c», «f» задают области, закрасненные соответствующим цветом. При запуске задания по ним формируются фигурки, из которых нужно складывать результат. Отметим тот факт, что область, заданная одной буквой, является цельной фигурой и поворачивается и перемещается синхронно, даже если не является связной.

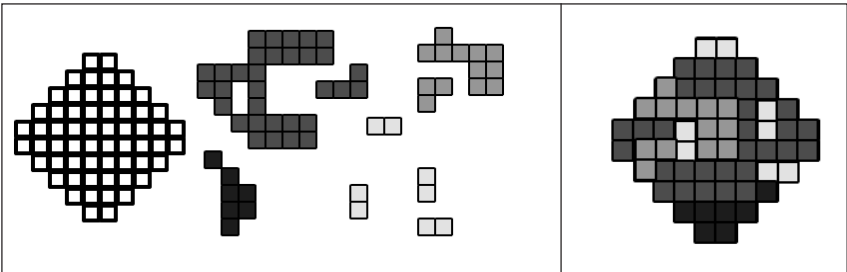


Рис. 4. Задание «Составить фигурку из компонент»

Координаты точек

На координатной плоскости нарисован многоугольник, необходимо указать координаты всех его вершин (рис. 5). Только в случае ввода правильного ответа, кружок напротив полей ввода становится зеленым, что сигнализирует ученику о том, что он правильно ввел координаты. Для формирования задания автор должен указать координаты вершин многоугольника в порядке обхода. Ниже приведен соответствующий текстовый файл для данного задания:

```
&x1=1&  
&y1=3&  
&x2=15&  
&y2=17&  
&x3=10&  
&y3=4&
```

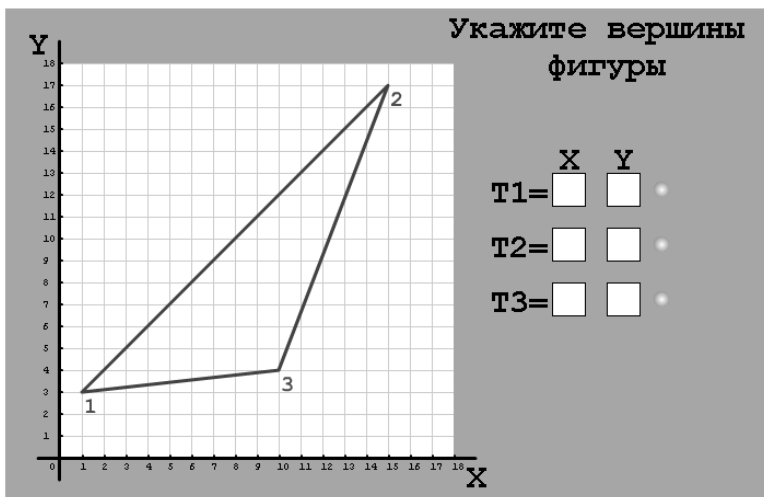


Рис. 5. Задание «Указать координаты точек»

Координаты точек, отрезки, длины

Это комбинированное задание. Вначале необходимо ввести координаты всех указанных на рисунке точек (рис. 6). За-

тем для всех обозначенных отрезков нужно указать номера точек, которые им соединяются. И, наконец, для каждого отрезка нужно указать его длину. Для отрезка, параллельного осям координат, довольно несложно подсчитать его длину в клеточках. Однако, если отрезок не параллелен осям координат, это уже может вызывать определенные сложности, тем более, у учеников начальной школы. Поэтому задание содержит подсказку. Достаточно навести курсор мыши на букву – название отрезка, и появится «хинт» – длина данного отрезка.

Для формирования задания автор указывает координаты всех точек, как это было показано для предыдущего задания, а также указывает, какие отрезки соединяют какие точки, как показано ниже:

```
&kolOtr=10&
&otr1=1 3&
&otr2=2 4& ...
```

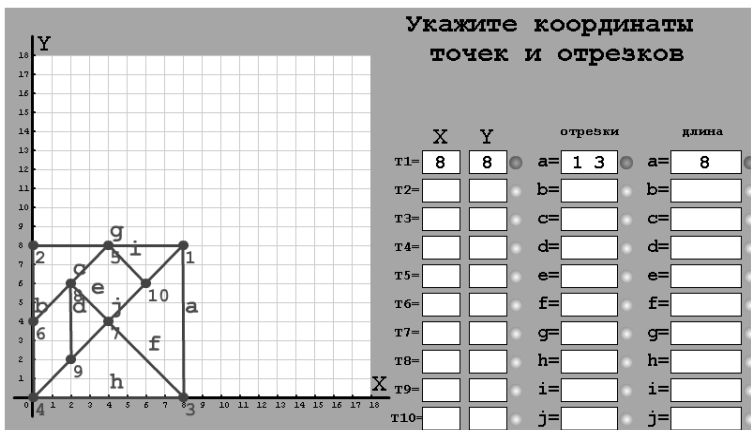


Рис. 6. Задание «Координаты точек, отрезки, длины»

Координаты, отрезки (исчезновение)

Данное задание готовится автором и выполняется учеником аналогично предыдущему (рис. 7). Важным отличием является исчезновение правильно введенных отрезков, а после ввода всех отрезков – координат точек.

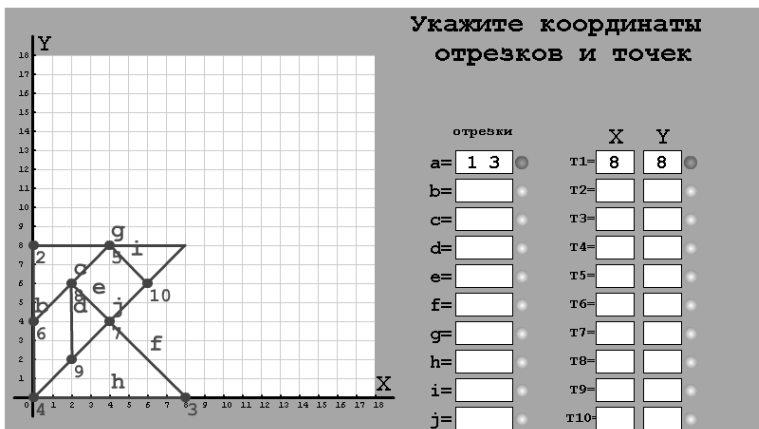


Рис. 7. Задание «Координаты, отрезки (исчезновение)»

Такой способ отображения правильности выполнения задания вызывает дополнительный неподдельный интерес у учеников.

Отрицательные координаты, отрезки, длины

Данный шаблон, поддерживая все ранее описанные функции, имеет дополнительные возможности по формированию задания, в том числе: отрицательные координаты и изменение размеров клетки (рис. 8). Для этого задаются величины \min и \max , обозначающие минимальное и максимальное из значений по обоим осям. Если $\min < 0$ тогда появится поле с отрицательными координатами, иначе ($\min \geq 0$) поле будет только с положительными координатами).

Для варианта $\min < 0$:

- если $\max \leq 4$ максимальное значение на осях координат будет 4;
- если $4 < \max \leq 6$ максимальное значение на осях координат будет 6;
- если $6 < \max \leq 9$ максимальное значение на осях координат будет 9.

Для варианта $\min \geq 0$:

- если $\max \leq 6$ максимальное значение на осях координат будет 6;

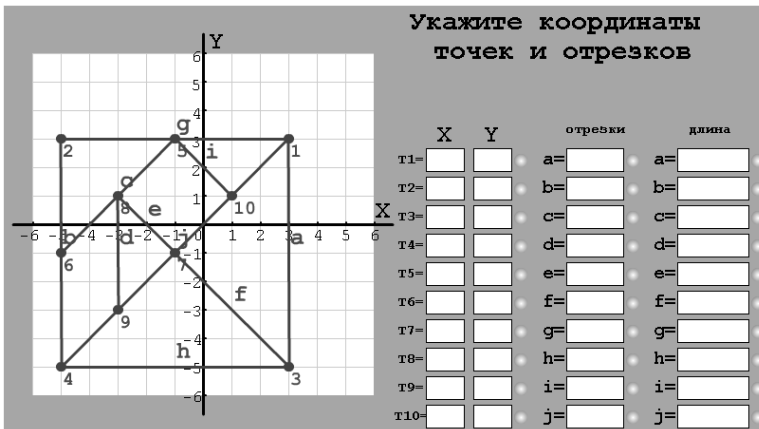


Рис. 8. Задание «Отрицательные координаты»

- если $6 < \max \leq 12$ максимальное значение на осях координат будет 12;
- если $12 < \max \leq 18$ максимальное значение на осях координат будет 18.

Указание позиции

При выполнении задания в левой области в одном из квадратиков (выбираемом случайно) появляется знак вопроса (рис. 9). Ученик должен кликнуть в квадратике, который нахо-

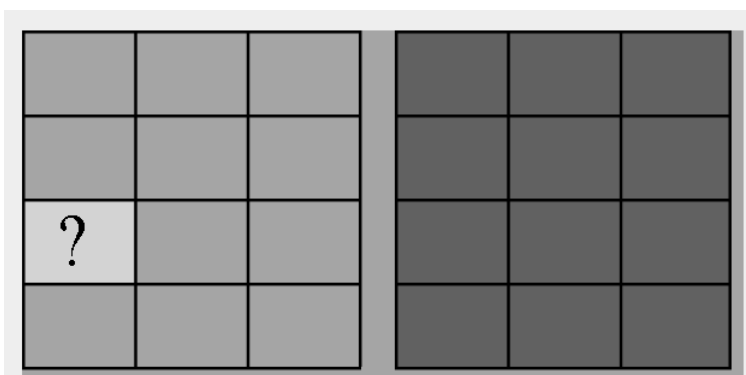


Рис. 9. Задание «Указать позицию»

дится на такой же позиции в области справа. Если ученик кликнул в правильном поле, оно перекрашивается с синего цвета на зеленый. Если ученик кликнул не в том поле, где было нужно, цвет поля остается прежним, и на нем появляется красный крестик. Количество допустимых неверных ответов задается автором задания. В случае превышения допустимого предела ошибок задание переводится в начальное состояние.

Параметрами задания, задаваемыми автором, являются также количество строк и столбцов в обоих полях, и то, в каком поле будут появляться вопросы – в правом или в левом.

Пропедевтика аналогии

В левой части задания представлен образец (набор правил), по которым цифры перемещаются из верхнего прямоугольника в нижний (рис. 10). В правой части предложен новый вариант размещения цифр в верхнем прямоугольнике. Ученик должен понять предлагаемый набор правил по образцу и расставить правильно цифры в нижнем прямоугольнике. При этом данное задание «ведет за руку ученика». В начале вопросик появляется в очередном свободном поле правого нижнего прямоугольника. Ученик должен кликнуть на соответствующем поле левого нижнего прямоугольника. После этого

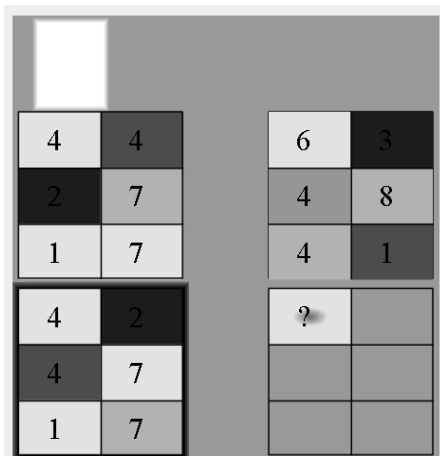


Рис. 10. Задание «Расставь цифры»

В последние автоматически включаются все правильные элементы ответа и добавляются автором задания дополнительные неправильные, которые случайно перемешиваются.

Звездочки в правой части задания показывают, сколько ошибок может допустить ученик при перетаскивании элементов (это также является параметром шаблона).

Найти ошибки в программе

Данное задание построено по принципу нахождения отличий между двумя рисунками (рис. 12). В качестве рисунков при этом выступают два текста программ, в одном из которых сделаны ошибки. Ученик должен кликами указать позиции ошибок.

Количество неверных кликов может быть ограничено автором задания, после чего приходится выполнять задание сначала. Кнопка «Сначала» может быть использована для того, чтобы ученик мог повторить самостоятельно выполнение задания, если ему была оказана помощь. Ниже приведен фрагмент текстового файла задания (символами «~») задаются отличия:

```
&R01=program ~num;~&
```

```
&R02=var&
```

```
&R03= s : ~string;~&
```

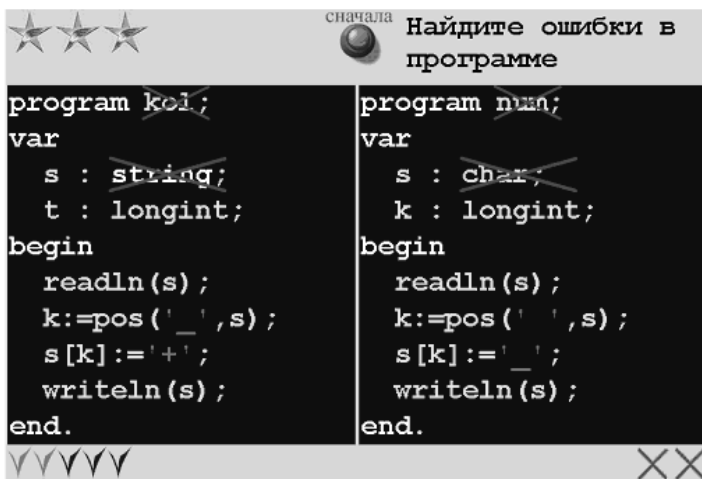


Рис. 12. Задание «Найти ошибки в программе»

Исправить ошибки в программе

Ученик должен, сравнивая построчно образец, представленный слева, с текстом для исправления, представленным справа, находить отличия и исправлять их аналогично тому, как это делается в среде «Turbo Pascal» (рис. 13). Если строка свободна от ошибок, то белая точка напротив этой строки меняется на зеленую.

Тексты программ задаются аналогично предыдущему заданию. Переменные R01, R02, ... задают правую (с ошибками) программу, а переменные L01, L02, ... соответствующие строки левой (эталонной) программы. Знаками «~» автор задания отмечает отличия программ.

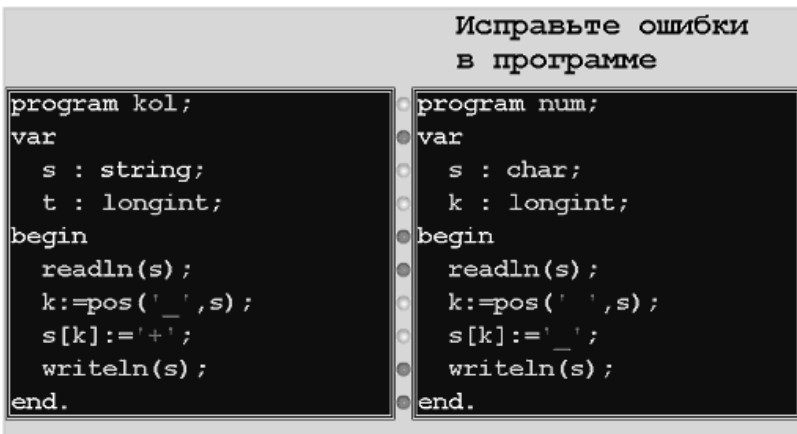


Рис. 13. Задание «Исправить ошибки в программе»

Набрать программу по образцу

Важным достоинством этого задания для начинающих, является возможность построчного контроля за корректностью набора программы и интерактивная реакция практически сразу после набора очередной строки. В то же время текст набираемой программы задается в файле параметров задания аналогично тому, как это делалось в предыдущих заданиях (рис. 14).

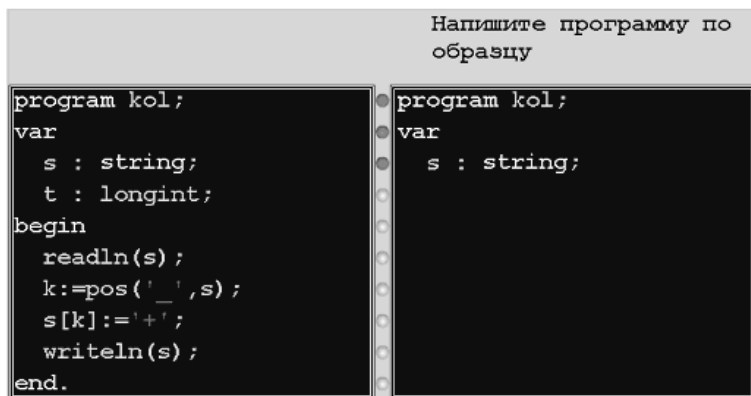


Рис. 14. Задание «Набрать программу по образцу»

Заключение

В данной статье описаны флеш-шаблоны, которые используются для быстрого создания интересных и полезных заданий развивающего обучения в системе DL. Отмечено, что сами шаблоны и примеры их применения находятся в открытом доступе и могут быть использованы для создания собственных курсов развивающего обучения, как на локальных машинах пользователей, так и загружая их в систему DL. Необходимо также отметить, что значительный обучающий эффект имеет и предложение школьникам (в том числе и младшим школьникам) самостоятельно придумывать и создавать новые задания и загружать их в систему DL для всеобщего пользования. Авторы с благодарностью воспримут как конструктивную критику, так и предложения по развитию системы флеш-шаблонов.

Необходимо отметить, что флеш-шаблоны – не единственное средство повышения производительности авторов учебных курсов, но эти другие средства будут освещаться в других статьях.

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. Развивающее дифференцированное обучение программированию в начальной школе / М.С. Долинский, М.А. Кугейко // Печатковая школа. – 2011. – № 1(229). – С. 41-45.

7. 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.

8. 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.

Статья поступила 31.03.2011

