

ИЗ ОПЫТА РАБОТЫ

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

Создание электронных учебных пособий для вузовских дисциплин с помощью конструктора флеш-заданий

Введение

В серии публикаций [1-13] описана авторская система развивающего обучения программированию, ориентированная на подготовку школьников к олимпиадам по информатике от школьной до международной. Важной составляющей этой системы являются задания, которые выполнены на базе флеш-технологий. В работах [9-13] были описаны возможности специально разработанного «Конструктора» флеш-заданий для развивающего обучения. Использование заданий, созданных таким образом, для фронтального обучения младших школьников СШ № 27 г. Гомеля и факультативного обучения на базе сайта <http://dl.gsu.by> (далее DL) школьников разных возрастов г. Гомеля и Гомельской области, показало высокую эффек-

тивность такого подхода. Более того, создание и использование заданий в «Конструкторе», с одной стороны, стимулировало его развитие, а с другой наглядно показало, что его возможности стали значительно шире, чем задумывалось изначально. В результате непрерывной апробации и развития «Конструктор» вырос в полноценную открытую среду для создания современных интерактивных электронных средств обучения. Данная статья посвящена созданию с помощью «Конструктора» интерактивных электронных учебных пособий для вузовских дисциплин.

На сегодняшний день интерактивные задания с помощью такого подхода создавались нами для следующих вузовских предметов: «Программирование», «Организация и функционирование ЭВМ», «Архитектура вычислительных систем», «Проектирование аппаратно-программных вычислительных средств».

Пример создания электронного учебного пособия

Далее приводятся пример использования данной технологии для построения комплекса обучающих заданий по теме «Проектирование цифровых устройств, заданных описаниями или таблицами истинности с помощью минимизации логических функций картами Карно»:

Например, пусть имеются два задания на разработку устройств:

1. Спроектируйте схему, которая работает в соответствии таблицей истинности, представленной в таблице 1.

2. На входе дана последовательность из 4 бит. На выход подать сначала все нули исходной последовательности, затем единицы.

Во втором случае вначале требуется составить таблицу истинности по описанию задания.

Таким образом, в общем случае, процесс выполнения задания можно разбить на несколько этапов:

1. Составить таблицу истинности по описанию задания.

2. Для каждого выхода внести единички в соответствующую карту Карно.

Таблица 1

x1	x2	x3	x4	y1	y2	y3	y4
0	0	0	0	0	0	1	0
0	0	0	1	1	0	0	0
0	0	1	0	1	0	0	0
0	0	1	1	0	0	1	0
0	1	0	0	1	0	0	0
0	1	0	1	0	0	1	0
0	1	1	0	0	0	1	0
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	0
1	0	0	1	0	0	1	0
1	0	1	0	0	0	1	0
1	0	1	1	1	0	0	0
1	1	0	0	0	0	1	0
1	1	0	1	1	0	0	0
1	1	1	0	1	0	0	0
1	1	1	1	0	0	1	0

3. Покрыть все единички минимальным количеством кораблей (с площадью степенью числа 2) максимальной площади.

4. Выписать логическую функцию для выполненного покрытия.

5. Нарисовать схему в системе проектирования HLCCAD [14].

На лекции, посвященной данной теме (специальность – Программное обеспечение, курс – 1, предмет – «Организация и функционирование ЭВМ»), вначале всем рассказывается, как решать поставленные задачи, а потом предлагается в командах решать такие задачи. Студенты используют личные ноутбуки, по одному, как минимум, на команду из двух человек, как для работы в HLCCAD, так и для отсылки решений на автоматическую проверку в систему DL. На проекторе все время отображаются текущие результаты – какая команда сколько задач решила. В названии команды присутствуют название группы и фамилии членов команды.

Все компьютеры и студентов, и преподавателя находятся в сети университета посредством беспроводного соединения.

Однако некоторым студентам, а иногда и командам, недостаточно общего пояснения. В этот момент на помощь приходят описываемые ниже обучающие задания. В них полное задание разбивается на «микроскопические» шаги, которые по отдельности легче выполняются. Кроме того, на каждый из этих шагов имеется подсказка в электронной версии только что прочитанной лекции. Часто бывает, что пройдя данное обучение, команды успевают еще на лекции вернуться к выполнению контрольных заданий. Кроме того, эти учебные задания могут использовать при самостоятельной подготовке студенты, по какой-то причине пропустившие лекцию, или желающие в индивидуальном порядке закрепить пройденный материал.

На рис. 1 представлено задание, в котором студент должен с помощью кликов мышки по перечисляемым полям (содержащим для выбора ?, 0, 1) установить систематический перечень всех возможных входных значений на данном устройстве (определить область определения булевой функции от четырех переменных x_1, x_2, x_3, x_4). Первые три строки в начале задания также содержали вопросы.

На рис. 2 представлено задание, в котором студент должен с помощью кликов мышки по перечисляемым полям (содержащим для выбора ?, x_1, x_2, x_3, x_4) выбрать наименования аргументов карты Карно (названия переменных).

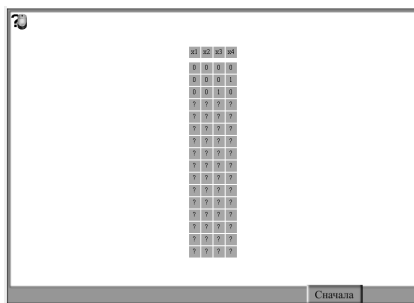


Рис.1 Область определения функции

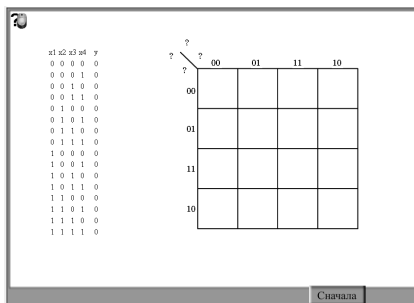


Рис.2. Названия переменных

На рис. 3 представлено задание, в котором студент должен с помощью кликов мышки по перечисляемым полям (содержащим для выбора ?, 00, 01, 10, 11) выбрать наименования строк и столбцов карты Карно.

На рис. 4 представлено задание, в котором студент должен с помощью кликов мышки по перечисляемым полям (содержащим для выбора ?, « » (пробел), 1) выбрать значения клеток карты Карно. Каждый знак вопроса должен быть заменен на 1 или пробел (вместо 0, для лучшей читабельности). Такое задание дается для каждой выходной переменной.

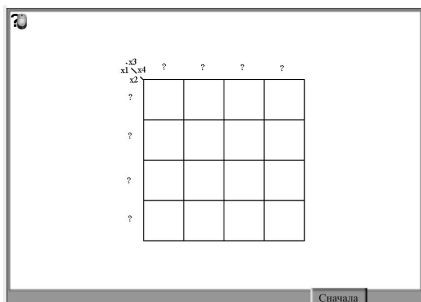


Рис. 3. Обозначения строк и столбцов

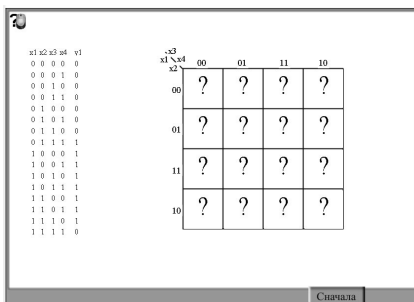


Рис. 4. Внесение единиц в карту Карно

На рис. 5 представлено задание, в котором студент должен покрыть все единицы минимальным количеством имеющихся кораблей максимальной площади. Корабль не может покрывать пустые поля (поля, в которых функция

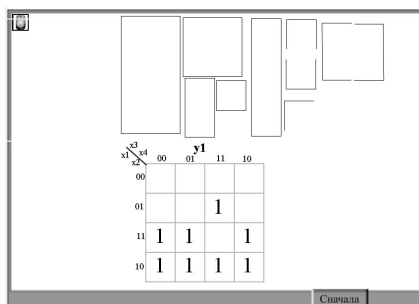


Рис. 5. Покрытие единиц на карте Карно

имеет значение 0). Все корабли имеют площадь, равную степени двойки. Они представлены в верхней области задания. Студент может мышкой указать требуемый корабль, перенести его в требуемую область, повернуть, если нужно и накрыть область из одних единичек. Отметим, что имеются корабли (представленные двумя половинками), которые поддерживают покрытие смежных крайних строк (верхнюю вместе с нижней или левую вместе с правой). Таких заданий также предлагается четыре – по числу выходных переменных.

На рис. 6 представлено задание, в котором по выполненным покрытиям требуется написать логические функции. В правом верхнем углу задания приводится пример выполнения такого задания. Для упрощения задания студентам вначале предлагается выписать логическую функцию отдельно по термам (использованным при покрытии кораблями). При вводе верных ответов текст подсвечивается зеленым цветом, пока он не содержит ошибок. При вводе ошибочных символов они сразу подсвечиваются красным цветом. В последнем поле ввода предлагается ввести полную логическую функцию выхода.

На рис. 7 представлено задание, в котором по имеющимся логическим функциям требуется составить функциональную схему устройства из конструктивных элементов системы проектирования HLCCAD, перетаскивая их мышкой на нужную позицию в схеме.

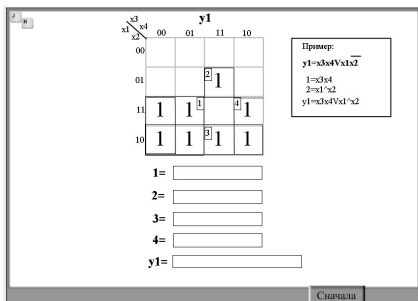


Рис. 6. Логические функции для выходных переменных

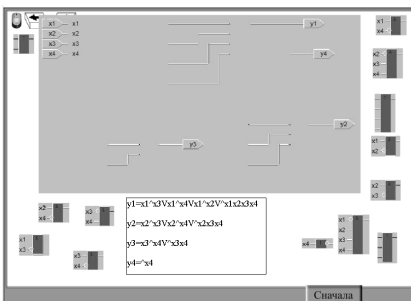


Рис. 7. Составить схему по логическим функциям

ся определенная подсказка студентам: если корпус поднесен на свою позицию, то он «прыгает» на нее (с расстояния в несколько пикселей – задается автором задания) и больше не перемещается.

На рис. 8 представлено задание, в котором студент должен составить схему по логическим функциям, как это делается с обычными пазлами, перетаскивая мышкой в нужное место фрагменты схемы.

На рис. 9 представлено задание, в котором студент должен для заданных входных значений, просчитать значения на всех контактах схемы и выходных, и промежуточных и указать эти значения с помощью выбора в перечисляемых полях, содержащих ?, 0, 1 .

Практика применения подобного обучения показала, что предложенному методу решения задач обучается 100% студентов, хотя время обучения может сильно варьироваться.

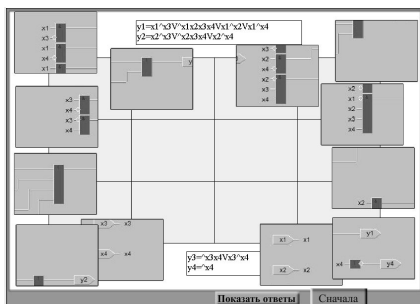


Рис. 8 Составить схему по логическим функциям (пазл)

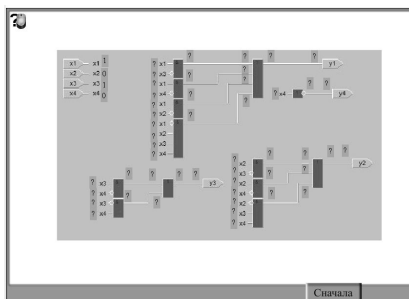


Рис. 9 Просчитать значения на всех контактах схемы

Другие примеры применения конструктора

В общем случае задания конструируются из активных и пассивных элементов. К активным элементам заданий относятся такие, которые позволяют обучаемому выполнять некоторые действия мышью и/или клавишами «стрелки», одновременно определяя, правильно ли выполнено задание. Пассивные элементы предназначены для

красочного и информативного оформления заданий. Активные элементы на момент написания статьи включают в себя таны, перечисляемые поля, области выделения (связные и независимые), области внесения, точки соединения, таймер, поля ввода.

Таны – это графические или текстовые элементы, которые при выполнении задания студент может перемещать с помощью мыши и поворачивать с помощью клавиш (стрелки влево и вправо). Таны делятся на графические и текстовые. Графические таны делятся на стандартные (с параметрами, поставляются в конструкторе), пользовательские (нарисованные авторами заданий средствами конструктора), произвольные рисунки (jpeg, png, разрешение не больше 742x500, размер не более 150 Кбайт). Автор задания может создавать собственные графические таны встроенными средствами конструктора (отрезками линий и фрагментами сплайнов второго порядка).

Текстовые таны создаются пользователем в конструкторе набором соответствующих текстов и выбором цвета текста и фона, а также шрифта и размера текста. В целях сокращения времени на создание заданий, обеспечена возможность использовать ранее созданные ресурсы: база заданий, база пользовательских танов (из всех заданий базы), библиотека пользовательских танов-рисунков (тематически сгруппированных).

Кроме танов имеются также следующие активные элементы заданий:

Перечисляемые поля – области, для которых автор задания указывает возможные варианты ответов из которых последовательными кликами студент выбирает правильный, по его мнению, ответ. Области выделения – клики в которых означают правильный ответ. Области внесения – для правильного выполнения задания достаточно внести тан в любое место этой области.

Точки соединения – для правильного выполнения задания требуется кликнуть мышью в некоторых из них. Таны, перечисляемые поля и поля ввода встречались в представленных выше заданиях.

Приведем примеры заданий, в которых используются еще не проиллюстрированные конструктивные элементы.

Эти задания использовались при изучении темы «Введение в ассемблер Intel 8086».

На рис. 10 представлены области внесения: название регистра достаточно внести в нужную ячейку сетки.

На рис. 11 представлено задание, использующее точки соединения – необходимо соединить фрагменты программ с ответами, которые они получают для выходной переменной r для заданных значений входных переменных x и y.

На рис.12 представлено задание, где используются области выделения. Студент обязан кликнуть в тех местах программы, где совершена ошибка.



Рис. 10 Область внесения

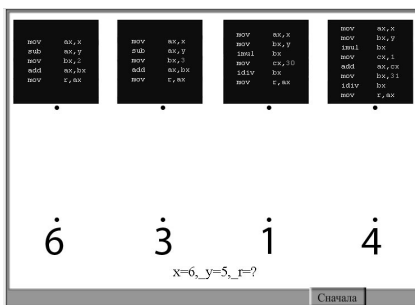


Рис. 11 Точки соединения

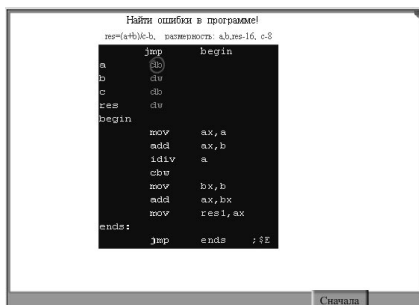


Рис.12 Области выделения

Заключение

Описанные в данной работе задания активно используются в учебном процессе математического факультета Гомельского государственного университета им. Ф. Скорины и обеспечивают существенную интенсификацию и повышение качества обучения по соответствующим дисциплинам.

Задания, созданные в описанном в данной работе конструкторе заданий, использовались авторами в широком спектре применения от вуза до дошкольного обучения. Как и предполагалось, задания в силу своей интерактивности и красочности привлекательны для обучаемых, а в силу мощности возможностей конструктора имеют высокую эффективность в обучении. Важными достоинствами конструктора являются простота использования и существенное снижение трудоемкости создания заданий.

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. Редактирование, симуляция и отладка аппаратного обеспечения с помощью HLCCAD / М.С. Долинский, И.В. Коршунов// Электроника-инфо. – 2010. – № 6(75). – С. 22-26.

Статья поступила 29.10.2012

