

## СИСТЕМА ТЕСТИРОВАНИЯ ПО СПЕЦИАЛЬНОЙ ДИСЦИПЛИНЕ В ТЕХНИЧЕСКОМ ВУЗЕ

*И.А. Кривиченко, Б.В. Никульшин*

*Белорусский государственный университет информатики и радиоэлектроники, Минск, Беларусь, iarifina@mail.ru*

Abstract. This article describes the main problems in the development of testing systems, as well as the principles of task control.

В процессе обучения при работе с системой тестирования студенты, как правило, стараются пройти все возможные варианты тестовых заданий, чтобы при итоговом контроле все задания были знакомы им. При дистанционной форме обучения у студентов больше возможностей сделать это. Поэтому при разработке системы тестирования желательно решить проблему ограниченной вариативности.

В системе тестирования по курсу САиИО основной упор делается на тестирование практических заданий. В данной системе предлагается следующий алгоритм решения проблемы ограниченной вариативности:

- ввод варианта,
- генерация первого числа и выбор шаблона задания из базы данных,
- генерация массива чисел и заполнение ими шаблона задания,
- выполнение задания студентом.

Остановимся на каждом этапе более подробно.

1. Сначала студенту предлагается ввести вариант, выданный ему преподавателем. Следует отметить, что при использовании данного алгоритма номер варианта может принимать любые натуральные значения в пределах от 1 до  $2^{32}$  (4 294 967 296) для ПК с 32-разрядной ОС и до  $2^{64}$  (8 589 934 592) для ПК с 64-разрядной ОС.

2. Далее, используя алгоритм генерации случайных чисел, генерируется первое число, при этом введенный номер варианта используется как один из входных параметров используемого алгоритма генерации случайных чисел.

3. Сгенерированное число умножается на количество шаблонов в БД и округляется до целого. В результате будет получен номер шаблона задания, который следует использовать в качестве основы тестового задания.

4. На следующем шаге можно генерировать массив чисел и, используя их, заполнять шаблон задания значениями числовых величин. После этого уже можно выводить задание на устройство ввода вывода. Студент может приступить к выполнению задания.

Данный алгоритм может быть применен к любой практической задаче. Желательно, сформулировать шаблоны задания таким образом, чтобы решения задания во многом зависело от числовых величин.

В результате количество вариантов заданий будет зависеть от двух факторов: используемого алгоритма генерации случайных чисел и его входных параметров и разрядности ОС.

При разработке системы был рассмотрен ряд алгоритмов генерации случайных равномерно распределенных чисел. В качестве наиболее важных характеристик рассматривались равномерность распределения первого элемента последовательности при разных значениях входного параметра, т.е. варианта, что обеспечивает равномерность из пользования БД и равномерность распределения 20 последовательных элементов при разных значениях входного параметра, что обеспечивает различие двух вариантов, заданных одним шаблоном.

Наиболее оптимальным является алгоритм Лемера [1]. Рекуррентная формула в алгоритме Лемера выглядит следующим образом:

$$x_n = (ax_{n-1} + c) \bmod m. \quad (1)$$

На параметры  $a$ ,  $c$ ,  $m$  накладываются следующие ограничения:

1.  $c$  и  $m$  взаимно просты,
2.  $(a - 1)$  кратно всем простым делителям  $m$ .

Значения первого члена  $x_0$  следует взять равным введенному номеру варианта.

Еще одним нестандартным решением в системе тестирования по курсу САиЮ стало разделение режимов обучения и аттестации.

В режиме обучения организована возможность просмотра студентом подробного описания совершаемых им ошибок, учитывая нюансы алгоритмов решения.

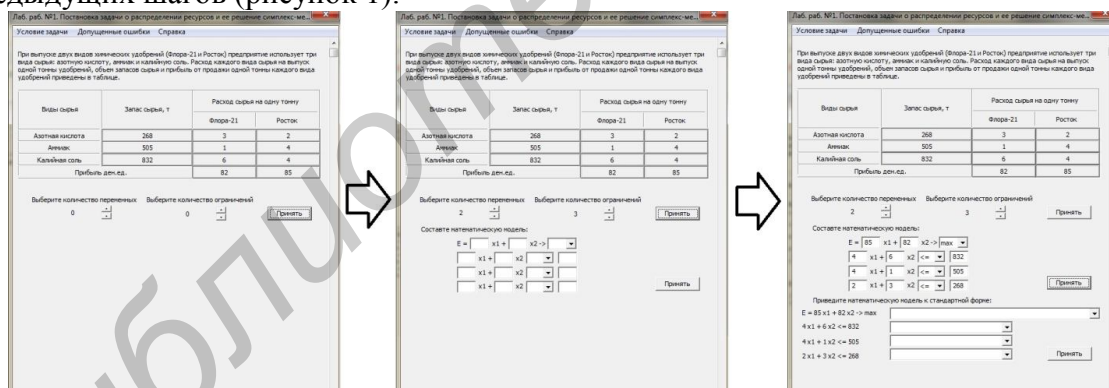
Аттестационный режим предназначен для проверки знаний студента. В этом режиме была организована отправка отчета о выполнении студентом задания тьютору.

Для того, чтобы полностью исключить совпадения заданий в аттестационном режиме и режиме обучения, были присвоены разные значения параметрам  $a$ ,  $c$ ,  $m$ .

Тестовые задание в системе тестирования представляют собой сложные задачи с решением в несколько шагов, поэтому проверка выполнения задания производится с пошаговым контролем правильности решения. Причем в системе тестирования есть два типа подобных задач:

1. задачи с решением в несколько шагов, где алгоритм гибок и зависит от формулировки конкретного задания;
2. задачи с решением в несколько шагов, где алгоритм решения задан четко и последующий шаг дает полный либо частичный ответ на предыдущий.

Для первого типа заданий используется модуль с визуализацией всех шагов, но доступным заполнением только текущего шага, что позволят студенту ориентироваться в ходе решения теста и видеть требуемое от него конечное решение задания. Для второго типа заданий используется модуль с визуализацией, только текущего и предыдущих шагов (рисунк 1).



**Рисунок 1 - Модуль с визуализацией текущего и предыдущих шагов тестового задания**

Таким образом в системе тестирования по курсу САиЮ решена проблема ограниченной вариативности. Система имеет два режима тестирования, а ее тестовые модули адаптированы к алгоритмам заданий.

### Литература

1. Смородинский С.С., Батин Н.В. Оптимизация решений на основе компьютерных имитационных методов и моделей. Учебное пособие по курсу “Моделирование систем” для студентов специальности АСОИ дневной и дистанционной форм обучения. В 2-х частях. Часть 1. Минск, БГУИР, 2004. – 80 с.
2. Кривиченко И.А., Никульшин Б.В. «Разработка современных систем педагогического тестирования» - тезисы Международной научной конференции «Информационные технологии и системы».