

# МОДЕЛИ ГЕНЕРАТОРОВ ВОПРОСОВ ДЛЯ КОМПЬЮТЕРНОГО КОНТРОЛЯ ЗНАНИЙ

В.В. Кручинин, Ю.В. Морозова

Томский университет систем управления и радиоэлектроники

В данной статье представлены модели и алгоритмы генерации вопросов, которые могут быть использованы как для получения вопросов для тестового контроля знаний, так и для создания программных генераторов. Подробно рассмотрены шаблоны для вопросов типа меню. Предложена генерация вопросов для языка предметной области на основе словаря. Описаны методы генерации, основанные на таблицах. Показаны конкретные примеры шаблонов генераторов задач и их алгоритмы генерации для различных дисциплин. Описанные модели и алгоритмы реализованы в конкретных генераторах и внедрены в реальный учебный процесс в Томском межвузовском центре дистанционного образования.

The article covers basic models and algorithms to generate test questions which can be used to produce questions for knowledge test control, as well as to create program generators. Some menu template samples herein are considered in detail. It's by the way offered to generate test questions as applied to the subject domain language based on the «dictionary» as a new building block. Generation methods based on usage of tables are described. Specific generator template examples of problems and their generation algorithms are shown as applied to various disciplines. Models and algorithms of generating programs mentioned have already been successfully implemented into the educational process for today in Tomsk interuniversity center of distance education, TICDE.

Компьютерный контроль знаний является необходимой составляющей дистанционной технологии обучения. Практически все организации, занимающиеся дистанционным обучением, используют тестовые компьютерные программы [7]. Однако опыт показывает, что такие тесто-

вые программы имеют один существенный недостаток - малый объем тестовых вопросов [3]. Это приводит к тому, что студенты достаточно быстро создают банк ответов к таким экзаменам, что существенно снижает эффективность проведения компьютерного тестирования.

Выходом их создавшейся проблемной ситуации может служить использование генераторов вопросов - специальных программ, которые, используя формализованное представление знаний данной предметной области, синтезируют вопросы для компьютерного теста [1, 5].

Перечислим основные свойства генератора вопросов [4]: мощность множества генерируемых вопросов, управляемость последовательности генераций вопросов, сложность реализации. Все эти свойства существенно зависят от моделей предметной области, используемых для построения генератора.

Важно также отметить: использование датчика случайных чисел для генерации вопросов нежелательно, так как при достаточно большой мощности генератора можно получить одинаковые вопросы. Поэтому предлагается разрабатывать алгоритмы генерации, основанные на перечислении вопросов.

Ниже описаны достаточно простые модели и алгоритмы генерации вопросов, которые могут быть использованы как для получения вопросов для тестового контроля знаний, так и для создания программных генераторов.

### 1. Шаблоны для вопросов типа меню

Рассмотрим построение генераторов для меню вопросов. В таком генераторе предполагается, что заранее заготовлены формулировка вопроса и два множества: правильных и

неправильных вариантов. Важно, что эти два множества достаточно велики. Тогда конкретное меню вопроса можно построить следующим образом:

- 1) выдается формулировка вопроса;
- 2) из множества правильных вариантов выбирается подмножество;
- 3) из множества неправильных вариантов выбирается подмножество;
- 4) все записывается в конкретный вопрос и задается студенту.

Тогда, если множество правильных вариантов имеет 5 вариантов, множество неправильных имеет 6 вариантов, то общее количество вопросов, в случае, когда из каждого из множеств выбирается по два варианта, будет равно

$$n_{\text{var}} = C_5^2 C_6^2 = 150,$$

где  $C_m^k$  - число сочетаний из  $m$  элементов по  $k$ .

Как видно из формулы, количество вариантов может быть достаточно велико. Общая схема построения такого типа вопроса может быть следующая:

- 1) производится описание некоторой исходной ситуации;
- 2) записывается множество правильных утверждений относительно этой ситуации;
- 3) записывается множество неправильных утверждений относительно этой ситуации.

Рассмотрим пример [2].

*Отметить команды микроконтроллера МК51, при трансляции которых используется прямой адрес байта:*

Список верных ответов:

```
PUSH PSW
POP DPH
MOV SBUF, A
DJNZ P1, M1
XRL B, #20
INC TH0
ORL TMOD, #0FH
MOV B, @R0
ANL 20, A
CJNE A, SP, M2
```

Список неверных ответов:

```
MOV R5, A
CLR C
CLR 20
ANL C, 20
MUL AB
DIV AB
INC DPTR
LCALL TIME
MOV DPTR, #1000
AJMP BEGIN
```

В условие задачи вставляется 10 альтернатив (пять верных и пять неверных), которые случайным образом выбираются из предложенных списков, причем случайным образом из-

меняется и последовательность записи альтернатив в задании. Тогда общее число вариантов данного типа вопроса может быть  $C_{12}^5 \cdot C_{12}^5 = 627294$

## 2. Генерация вопросов для языка предметной области

Владение понятийным аппаратом изучаемой дисциплины является важнейшим элементом знаний студента. Обычно для описания понятийного аппарата предметной области используются толковые словари. Словарь представляется набором статей, каждая из которых содержит понятие и синонимы с кратким опре-

Термины	Определения
Термин 1	Определение 1
Термин 2	Определение 2
...	...
Термин n	Определение n

Используя эту таблицу, можно получить следующие типы вопросов:

### 1. Тип вопроса:

Укажите термин для следующего определения: «Определение  $i$ -го термина».

Варианты:

1. Термин  $i$ .
2. Термин  $j$ .
3. Термин  $k$ .
4. Термин  $l$ ,

где  $i, j, k, l$  - номера строк словаря, причем не равные между собой.

### 2. Тип вопроса:

Укажите определение для следующего термина: « $i$ -й термин».

делением этого понятия. Методы построения словарей можно найти в работе [6]. Ниже рассматривается словарь как модель генератора вопросов.

Для проверки владения понятийным аппаратом изучаемой дисциплины можно предложить следующий генератор вопросов.

Словарь записывается в виде следующей таблицы:

Варианты:

1. Определение  $i$ .
2. Определение  $j$ .
3. Определение  $k$ .
4. Определение  $l$ ,

где  $i, j, k, l$  - номера строк словаря, причем не равные между собой.

### 3. Тип вопроса:

Введите термин для следующего определения: «Определение  $i$ -го термина» . . .

## Пример 1

*Дан фрагмент словаря*

Байт	ячейка памяти размером в 8 бит
Слово	ячейка памяти, с которой оперирует аппаратная часть вычислительной машины
Регистр	ячейка памяти процессора
Адрес	номер ячейки памяти
Блок	фрагмент памяти, превышающий размер слова

Тогда можно получить следующие вопросы.

### 1. Вопрос первого типа.

Байт - это:

- 1) ячейка памяти процессора;
- 2) фрагмент памяти, превышающий размер слова;
- 3) ячейка памяти процессора;
- 4) ячейка памяти размером в 8 бит.

Укажите верный вариант.

Правильный ответ: четвертый вариант.

### 2. Вопрос второго типа.

Фрагмент памяти, превышающий размер слова, - это:

- 1) байт;
- 2) слово;
- 3) блок;
- 4) регистр.

Правильный ответ: третий вариант (блок).

3. Вопрос третьего типа.

Ячейка памяти, с которой оперирует аппаратная часть вычислительной машины, - это . . .

(Введите термин)

Правильный ответ: регистр.

### 2.1. Подсчет числа вариантов вопросов

Для подсчета количества вариантов для первого типа вопросов можно предложить следующую формулу:

$$v_1 = n C_{n-1}^{k-1},$$

где  $v_1$  - количество вопросов первого типа;  $n$  - количество статей в словаре,  $k$  - количество вариантов в меню.

Например, если  $n=10$ ,  $k=5$ , то общее число вариантов будет

$$v_1 = 10 \frac{9!}{4!5!} = 1260.$$

Для второго типа вопросов формула будет аналогичной:

$$v_2 = n C_{n-1}^{k-1}.$$

Для третьего типа вопросов можно сгенерировать только  $n$  вопросов. Тогда общее число вопросов, которое можно сгенерировать предложенным алгоритмом, будет следующее:

$$v = v_1 + v_2 + v_3 = 2n C_{n-1}^{k-1} + n.$$

### 2.2. Алгоритм генерации

Дано: словарь объемом в  $n$  словарных статей; некоторое число  $i$ , удовлетворяющее условию:

$$0 \leq i \leq v.$$

Необходимо: получить вопрос, и если это меню, то количество вариантов должно быть  $k$ .

1. Если  $i < v_1$ , то генерируем вопрос первого типа:

- вычисляем  $m$  - номер словарной статьи;
- вычисляем  $l=i/n$  - номер сочетания;
- генерируем  $l$ -сочетание  $(n-1, k-1)$ , используя алгоритм генерации сочетаний [4];

- записываем вопрос «Укажите термин для следующего определения», вставляем  $k$ -е определение;

- формируем вектор вариантов ответов, используя вектор сгенерированного сочетания и  $k$ -й термин. Затем перемешиваем вектор ответов и храним позицию правильного ответа. Вектор сочетания хранит номера словарных статей, вошедших в сочетание с данным номером

$l$ , при этом необходимо, чтобы  $k$ -я словарная статья не попала в сочетание, все номера, больше или равные  $k$ , увеличиваются на единицу.

2. Если  $i < v_1 + v_2$ , то генерируем вопрос второго типа:

- вычисляем  $k = i \% n$  - номер словарной статьи;
- вычисляем  $l = i / n$  - номер сочетания;
- генерируем  $l$ -сочетание  $(n-1, k-1)$ , используя алгоритм генерации сочетаний *GCombination* [4];

- записываем вопрос «Укажите определение для следующего термина», вставляем  $k$ -й термин;

- формируем вектор вариантов ответов, используя вектор сгенерированного сочетания и  $k$ -е определение. Затем перемешиваем вектор ответов и храним позицию правильного ответа. Вектор сочетания хранит номера словарных статей, вошедших в сочетание с данным номером  $l$ , при этом необходимо, чтобы  $k$ -я словарная статья не попала в сочетание, все номера, больше или равные  $k$ , увеличиваются на единицу.

3. Если  $i < v_1 + v_2 + v_3$ , то генерируем вопрос третьего типа:

- вычисляем  $k = v_1 + v_2 + v_3 - i$ , номер словарной статьи;
- записываем определение и формулируем вопрос;
- в эталонный ответ записываем термин.

### 3. Методы генерации, основанные на таблицах

Словарь является частным случаем табличного представления данных. Исследования, связанные с представлением таблиц в памяти компьютера, проводятся в рамках реляционной алгебры [6]. Ниже предлагается реляционная модель представления шаблона и алгоритм генерации вопросов.

Пусть имеется некоторая таблица, в которой систематизированы некоторые знания или факты из конкретной предметной области. Манипулируя полями такой таблицы, можно генерировать следующие вопросы:

- 1) меню;
- 2) заполнение пропущенного поля;
- 3) указание правильного варианта заполнения данного поля (меню вариантов);
- 4) расстановка перечисленных элементов в пропущенные поля таблицы (упорядочение).

Генерация меню вопроса по таблице:

1. Выбрать  $k$ -ю строку в таблице.
2. Выбрать 2 поля с номерами столбцов  $i$  и  $j$ , которые однозначно соответствуют друг другу.
3. Содержимое  $i$ -го поля поместить в формулировку вопроса, а содержимое  $j$ -го поля поместить как правильный вариант ответа.
4. Выбрать  $l$  полей в  $j$ -м столбце таблицы, причем номера строк соответствующих полей не должны равняться  $k$ .
5. Перемешать  $l$  выбранных полей с  $j$  полем.
6. Сформатировать вопрос.

### 3.1. Подсчет числа вариантов меню вопросов

Число вариантов меню вопросов можно вычислить по следующей формуле:

$$v = nm(m-1)C_{n-1}^l,$$

где  $n$  - число строк;

$m$  - число столбцов в матрице;

$m(m-1)$  - задает число размещений по два элемента, задает множество пар столбцов с учетом их следования;

$l$  - число неверных вариантов в меню вопроса;

$C_{n-1}^l$  - число сочетаний строк таблицы без выбранной строки, по  $l$ .

### 3.2. Алгоритм генерации меню вопросов

Пусть есть таблица [ $n$  строк,  $m$  столбцов], число неверных вариантов в вопросе  $l$  и задано некоторое целое число  $i$ , задающее некоторый искомым вопрос.

1. Найти номер строки:

$$- k = i \bmod(n);$$

$$- i = i/n.$$

2. Найти пару столбцов:

$$- R = i \bmod(m(m-1)),$$

- сгенерировать размещение  $A(2, m)$  [4];

$$- i = i/(m(m-1)).$$

3. Найти сочетание  $C_{n-1}^l(i)$  на множестве строк, в котором отсутствует  $k$ -я строка. Для этого необходимо использовать алгоритм генерации сочетания [4].

2. Объединить вектор сочетания со второй компонентой вектора размещения и перемешать.

3. Сформатировать вопрос.

Пример генератора вопросов на основе таблицы.

Пусть дана следующая таблица, описывающая периоды палеозойской эры [8]:

Пермь	280 млн лет назад	Исчезновение трилобитов и примитивных кораллов Расцвет звероподобных пресмыкающихся
Карбон	345 млн лет назад	Появление первых голосеменных растений Появление пресмыкающихся Первые крылатые насекомые Расцвет земноводных Первые леса
Девон	395 млн лет назад	Первые земноводные Развитие различных групп рыб Первые насекомые Первые древовидные растения Примитивная псилофитовая флора
Силур	445 млн лет назад	Появление челюстных рыб Появление наземных членистоногих Появление наземных растений Многочисленные коралловые рифы
Ордовик	500 млн лет назад	Расцвет головоногих Дальнейшее развитие членистоногих Эволюция кишечнополостных Расцвет граптолитов Бесчелюстные рыбы
Кембрий	570 млн лет назад	Появление иглокожих Появление трилобитов Развитие беспозвоночных Появление сосудистых растений Эволюция водорослей

Генерация меню вопросов:

**Пример 1**

Укажите название периода, который был 395 млн лет назад:

- 1) девон;
- 2) кембрий;
- 3) карбон;
- 4) пермь.

**Пример 2**

Укажите период, название которого карбон:

- 1) 280 млн лет назад;
- 2) 345 млн лет назад;
- 3) 500 млн лет назад;
- 4) 570 млн лет назад.

**Пример 3**

Какой период характеризуется следующим:

Первые земноводные
Развитие различных групп рыб
Первые насекомые
Первые древовидные растения
Примитивная псилофитовая флора

- 1) девон;
- 2) кембрий;
- 3) карбон;
- 4) пермь.

**Пример 4**

Какая группа характеристик соответствует периоду силур:

- 1) первые земноводные, развитие различных групп рыб,

- первые насекомые, первые древовидные растения, примитивная псилофитовая флора;
- 2) появление челюстных рыб, появление наземных членистоногих, появление наземных растений, многочисленные коралловые рифы;
- 3) расцвет головоногих, дальнейшее развитие членистоногих, эволюция кишечнорастных, расцвет граптолитов, бесчелюстные рыбы.

Введите соответствующий номер.

**Пример 5**

Какая группа характеристик соответствует периоду, который был 500 млн лет назад:

- 1) первые земноводные, развитие различных групп рыб, первые насекомые, первые древовидные растения, примитивная псилофитовая флора;
- 2) появление челюстных рыб, появление наземных членистоногих, появление наземных растений, многочисленные коралловые рифы;
- 3) расцвет головоногих, дальнейшее развитие членистоногих, эволюция кишечнорастных, расцвет граптолитов, бесчелюстные рыбы.

Введите соответствующий номер.

**3.3. Генерация с пропусками**

**Пример 1**

Введите значение в незаполненное поле (ячейку) следующей таблицы:

Пермь	Карбон	Девон	Силур
_____	345 млн лет назад	395 млн лет назад	445 млн лет назад

**Пример 2**

Введите значение в незаполненное поле (ячейку) следующей таблицы:

Пермь	_____	Девон	Силур
280 млн лет назад	345 млн лет назад	395 млн лет назад	445 млн лет назад

### 3.4. Генерация с использованием перемешивания

#### Пример 1

Дана таблица:

Пермь	Карбон	Девон	Силур
395 млн лет назад	345 млн лет назад	280 млн лет назад	445 млн лет назад

Расставьте периоды в соответствии с названиями.

#### Пример 2

Даны следующие периоды палеозойской эры:

- 1) карбон;
- 2) девон;
- 3) кембрий;
- 4) силур.

Записать периоды, начиная с самого раннего (позднего). В ответ ввести последовательность номеров, разделенных пробелами.

Предложенные модели и алгоритмы были использованы для построения генераторов вопросов в Томском межвузовском центре дистанционного образования (ТМЦДО). Было разработано и успешно эксплуатируется 22 генератора, в которых реализовано свыше 2000 разнообразных шаблонов и моделей. Каждый из генераторов получает уникальные вопросы для всех студентов ТМЦДО.

### ЛИТЕРАТУРА

1. Башмаков А.И., Башмаков И.А. Разработка компьютерных и обучающих систем. М.: Информационно-издательский дом «Филинь», 2003. 616 с.
2. Егоркина Ю.В., Кручинин В.В., Шарпов А.В. Пакет генераторов тестовых заданий по циклу «Цифровые микропроцессорные устройства» // Современное образование: Интеграция учебы, науки, производства: Материалы региональной научно-методической конф. Томск: Изд-во ТУСУР, 2003. С. 86-87.
3. Исакова О.Ю., Кручинин В.В. Основные направления совершенствования контроля знаний в ТМЦДО // Материалы 2-й Всероссийской научно-практической конференции «Единая образовательная среда: проблемы и пути развития». Томск: Изд-во Том. ун-та, 2003. С. 175-177.
4. Кручинин В.В. Генераторы в компьютерных учебных программах. Томск: Изд-во Том. ун-та, 2003. 200 с.
5. Кручинин В.В. Разработка компьютерных учебных программ. Томск: Изд-во Том. ун-та. 1998. 211 с.
6. Мейер Д. Теория реляционных баз данных. М.: Мир, 1987. 320 с.
7. Российский портал открытого образования: обучение, опыт, организация. / Отв. ред. В.И.Солдаткин. М.: МГИУ, 2003. 508 с.
8. Шпинар З.В. История жизни на земле. Прага: Артия, 1977. 228 с.