

Используя формулу (1) мы получим следующие нечеткие множества:

$$A_1 = \left\{ \frac{1}{x_1} + \frac{0,87}{x_2} + \frac{0,80}{x_3} + \frac{0,86}{x_4} + \frac{0,94}{x_5} \right\}, \quad A_2 = \left\{ \frac{1}{x_1} + \frac{0,65}{x_2} + \frac{0,81}{x_3} + \frac{0,66}{x_4} + \frac{0,99}{x_5} \right\},$$

$$A_3 = \left\{ \frac{0,41}{x_1} + \frac{0,63}{x_2} + \frac{0,51}{x_3} + \frac{0,54}{x_4} + \frac{1}{x_5} \right\},$$

где A_i – функция принадлежности (функция превосходства) степени сложности тем по каждому из 3-х критериев.

После формирования функций принадлежности необходимо определить относительную сложность тем. В соответствии с формулой (1), степень сложности тем с учетом степени важности критериев определяется следующим нечетким множеством:

$$D = \left\{ \frac{0,14}{x_1} + \frac{0,21}{x_2} + \frac{0,17}{x_3} + \frac{0,17}{x_4} + \frac{0,31}{x_5} \right\}.$$

После этого была сформирована нечеткая база знаний, которая представлена в таблице 2. Здесь a_1 и a_2 – коэффициенты сложности вопроса.

Таблица 2 – Нечеткая база знаний

№ темы	Идентификатор степени сложности темы	Количество баллов за правильный ответ	
		$a_1=0,3$	$a_2=0,7$
1.	$d_1=0,14$	0,95	1,23
2.	$d_2=0,21$	1,46	1,87
3.	$d_3=0,17$	1,17	1,5
4.	$d_4=0,17$	1,24	1,6
5.	$d_5=0,31$	2,18	2,8

Полученная база знаний включает в себя идентификатор сложности тем (чем выше идентификатор – тем сложнее тема), а также количество баллов за правильный ответ на вопрос определенной сложности. Чем сложнее вопрос, тем больше баллов получит студент за правильный ответ.

Заключение. Таким образом, решая задачу формирования правил обоснования количества баллов за одно тестовое задание на уровне структурной идентификации, была получена нечеткая база знаний, содержащая необходимую информацию для оценки сложности тем в стационарном тесте.

Литература:

1. Агаев, Ф.Т. Виртуальное образование и оценка сложности теста в обучающих системах / Ф.Т. Агаев // Междунар. науч.-технич. конф. «Информационные системы в экономике, управлении производством и образовании». – Т.1. – 2003. – С.10-13.
2. Штовба, С.Д. Введение в теорию нечетких множеств и нечеткую логику [Электронный ресурс] / Режим доступа: <http://matlab.exponenta.ru/fuzzylogic/book1/index.php>. – Дата доступа: 24.11.2016.

МОДЕЛИРОВАНИЕ ВИЗУАЛЬНОЙ СРЕДЫ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ ШКОЛЬНИКОВ

Юшков К.Д.,

студент 4 курса ВГУ имени П.М. Машерова, г. Витебск, Республика Беларусь

Научный руководитель – Потапова Л.Е., канд. физ.-мат. наук, доцент

Информатика рассматривается как важнейший компонент общего образования современного человека, играющий значимую роль в решении одной из приоритетных задач образования - формировании целостного мировоззрения, системно-информационной картины мира, учебных и коммуникативных навыков и основных психических качеств детей [1]. Практически все концепции реформирования общеобразовательной школы основываются на широком внедрении компьютерных технологий в учебный процесс. Владение хотя бы начальными навыками компьютерного моделирования и программирования значительно повышает эффективность использования компьютера. В связи с этим актуальны визуальные средства обучения программированию, позволяющие в игровой форме помочь школьникам освоить основные принципы алгоритмизации и программирования. В игре вырабатываются такие жизненно важные качества, как внимательность, усидчивость, память, упорство, настойчивость в достижении цели. Кроме того, игра развивает коммуникативные способности, логическое мышление и повышает интерес к деятельности [2]. Целью работы является моделирование визуальной среды обучения программированию для детей среднего школьного возраста.

Материал и методы. В исследовании в качестве рабочего материала использованы различные модели визуальных средств обучения программированию. Реализованы методы исследования общенаучного характера (анализ, синтез, обобщение, сравнение), наблюдение.

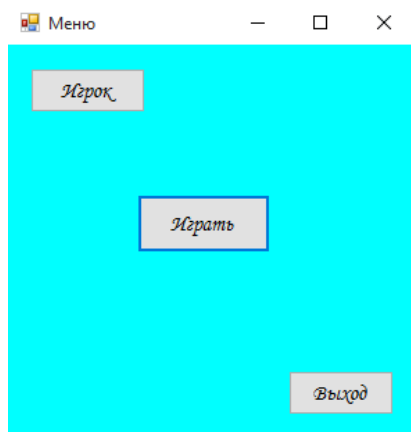


Рис. 1

Результаты и их обсуждение. В настоящей работе предложена модель визуальной игровой среды обучения программированию современному языку C# детей среднего школьного возраста. Она включает такие понятия как типы данных, ветвление, циклы и классы. Игра рассчитана на детей 6-8 класса.

Главное окно игры содержит меню выбора варианта действий (рис. 1)

При нажатии на какую-нибудь кнопку в главном окне, появляется дочернее окно. Если пользователь нажимает на кнопку *Игрок*, то создается новый игрок или вызывается существующий.

Логика игры предполагает наличие нескольких уровней, отражающих сложность изучаемых понятий. После выбора или создания игрока появляется окно, где игрок может нажать на кнопку *Играть*, которая вызовет создание новой формы для выбора уровня.

Первый уровень направлен на знакомство с типами языка программирования C# [3].

Второй уровень является закреплением первого. Школьнику нужно правильно расставить типы в зависимости от инициализации переменной. Этот уровень научит объявлять и инициализировать переменные, что является неотъемлемой частью программирования.

Третий уровень познакомит с ветвлениями. Например, ученику предоставлена миссия спасти «Стича» от дождя (рис. 2).

Четвертый и пятый уровни познакомят с циклом *while* и *for*. Решая предложенную задачу, ребенок поймет, как работать с циклами.



Рис. 2

Последний – шестой уровень, дает такие понятия как класс, свойства класса.

На всех уровнях нужно перетаскивать или выбирать правильные варианты ответа, это позволяет в доступной форме освоить довольно сложные структуры языка программирования.

Заключение. Разработанная визуальная среда позволяет в игровой форме обучать школьников программированию на языке C#. Удобный интерфейс игры, с одной

стороны, позволяет легко ориентироваться в игре, не требуя от пользователя каких-либо специальных навыков работы с электронно-вычислительными машинами, с другой стороны предоставляет игроку возможность освоить основы программирования на языке C#.

Литература:

1. Батрашина, Г.С. Игра как метод изучения моделей в начальной школе / Г.С. Батрашина // Информатика и образование. – 2008. – № 3. – С. 54–67.
2. Великович, Л. Программирование для начинающих / Л. Великович, М. Цветкова. – М.: Бином, 2007. – 258 с.
3. Singleton C# [Электронный ресурс].- Режим доступа: <http://metanit.com/sharp/patterns/2.3.php> . – Дата доступа: 24.12.2016.