



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ГУМАНИТАРНО-  
ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»

(ФГБОУ ВО «ЮУрГГПУ»)

ФИЗИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ

КАФЕДРА ИНФОРМАТИКИ, ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И  
МЕТОДИКИ ОБУЧЕНИЯ ИНФОРМАТИКЕ

Использование средств виртуальной реальности для поддержки изучения  
ветвлений в курсе программирования

Выпускная квалификационная работа

по направлению 44.03.01 Педагогическое образование

Направленность программы бакалавриата

«Информатика»

Выполнил:

Студент группы ЗФ-513-092-5-1

Гриневич Артём Дмитриевич

Паршукова Наталья Борисовна  
Физико-математический факультет,  
к.п.н., доцент кафедры ИИТиМОИ

Проверка на объем заимствований:

76,91 % авторского текста

Работа рекомендована к защите

рекомендована/не рекомендована

« 28 » мая 20 19 г.

зав. кафедрой И, ИТ и МОИ

Рузаков А.А.

Челябинск

2019



**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное бюджетное образовательное учреждение  
высшего образования**

**«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ГУМАНИТАРНО-  
ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «ЮУрГГПУ»)**

**ФИЗИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ**

**КАФЕДРА ИНФОРМАТИКИ, ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И  
МЕТОДИКИ ОБУЧЕНИЯ ИНФОРМАТИКЕ**

**Использование средств виртуальной реальности для поддержки изучения  
ветвлений в курсе программирования**

**Выпускная квалификационная работа**

**по направлению 44.03.01 Педагогическое образование**

**Направленность программы бакалавриата**

**«Информатика»**

Выполнил:

Проверка на объем заимствований:

\_\_\_\_\_ % авторского текста

Студент группы ЗФ-513-092-5-1

Гриневич Артём Дмитриевич

Работа \_\_\_\_\_ к защите

рекомендована/не рекомендована

« \_\_\_ » \_\_\_\_\_ 20\_\_ г.

зав. кафедрой И, ИТ и МОИ

Паршукова Наталья Борисовна

Физико-математический факультет,

к.п.н., доцент кафедры ИИТиМОИ

\_\_\_\_\_ Рузаков А.А.

**Челябинск**

**2019**

## ОГЛАВЛЕНИЕ

ОГЛАВЛЕНИЕ	2
ВВЕДЕНИЕ	3
ГЛАВА 1. ВЕТВЛЕНИЕ НА ЯЗЫКЕ PASCAL	6
1.1 Алгоритм ветвления и его реализация	6
1.2 Существующие ЦОРы для обучения ветвлению	10
1.3 Тестирование в образовательном процессе	12
1.4 Процесс обучения в виртуальной реальности	13
1.5 Выбор средств создания виртуальной реальности	14
1.6 Реализации ЦОР на движке Unity3d	15
Вывод по главе 1	21
ГЛАВА 2. Методика обучения по теме «Ветвление на языке Pascal» с использованием средств виртуальной реальности	22
2.1 Составление требований к цифровому обучающему ресурсу	22
2.2 Методика проведения урока «Ветвление»	23
2.3 Описания разработанного ЦОР «Ветвление»	25
2.4 Апробация результатов исследования	31
Вывод по главе 2	33
Заключение	34
Список литературы	35
Приложение 1	38
Приложение 2	50

## ВВЕДЕНИЕ

На уроках информатики, так или иначе, рассматривается тема построения алгоритмов решения задач. Алгоритм – это точное описание процесса выполнения операций или команд, который зачастую содержит инструкции — как надо выполнять ту или иную операцию. Техника составления алгоритмов для решения задач называется алгоритмизацией. В школьном курсе учеников знакомят с линейными и нелинейными алгоритмами. Линейные алгоритмы представляют собой однозначное и последовательное выполнение всех операций составляющих алгоритм решения задачи. Нелинейный алгоритм напротив — может иметь в себе условия, при которых выполняются, только часть операций. Основными случаями, когда необходимо применять ветвление:

- 1) когда часть команд или операций нужно пропускать в зависимости от условия;
- 2) когда алгоритм предусматривает выбор действий зависящих от условия.

В современном мире большинство задач решаемых с использованием компьютерных программ, нуждаются в операциях ветвления, будь то это проверка входных данных или развилки в бизнес кейсах. Соответственно умение реализовывать программы с ветвлением – одно из основных умений, которое должны получить учащиеся в школе.

**Противоречие:** между необходимостью качественного процесса обучения школьников ветвлению на языке Pascal и недостаточным методическим обеспечением данного процесса.

**Проблема исследования,** - каковы возможности средств виртуальной реальности для поддержки изучения ветвления на уроке информатики в рамках курса программирования.

**Цель исследования:** изучение возможностей средств виртуальной реальности для разработки цифрового образовательного ресурса полной поддержки урока «ветвление».

**Объект:** процесс обучения школьников теме «ветвления» на уроке информатики.

**Предмет исследования:** формирование умений использовать все способы записи условных операторов на языке Pascal.

**Основные задачи исследования:**

- 1) изучить существующие цифровые образовательные ресурсы по теме ветвление;
- 2) рассмотреть существующие материалы для методической поддержки проведения урока «ветвление»;
- 3) раскрыть роль возможности средств виртуальной реальности;
- 4) проанализировать результаты экспериментальной работы;
- 5) реализовать цифровой образовательный ресурс, использующий средства виртуальной реальности.

**Гипотеза исследования:** формирование знаний по теме ветвление у школьников будет более эффективным, если внедрить в урок цифровой образовательный ресурс с элементами виртуальной реальности. Элементы виртуальной реальности улучшает фокусировку внимания учащегося, что повышает качество процесса обучения.

Для решения поставленных задач и проверки гипотезы нами применялись следующие **методы исследования:**

- 1) теоретические методы: анализ исследований о влияние виртуальной реальности на образовательные процессы;
- 2) эмпирические методы: эксперимент.
- 3) методы качественной, количественной обработки результатов.

**Практическая значимость** работы заключается в том, что разработанный цифровой образовательный ресурс для поддержки темы

«ветвления» может быть использована в практической деятельности учителей 8 и 9 классов, педагогов дополнительного образования.

**Методологическая база:** работы педагогов, таких как: Поляков К. Ю., Еремин Е. А, Босова Л. Л., Семакин И.Г. Также использованы научные работы по следованию тестирования как образовательной деятельности: Раш Г., Лобанева А. Р. и др.

**База исследования:** ГБПОУ "Южно-Уральский государственный колледж"

**Структура выпускной квалификационной работы включает в себя:** оглавление, введение, 2 главы, заключение, список использованной литературы, приложение.

# ГЛАВА 1. ВЕТВЛЕНИЕ НА ЯЗЫКЕ PASCAL

## 1.1 Алгоритм ветвления и его реализация

Язык Pascal в настоящее время является наиболее популярным языком программирования высокого уровня, применяемым в общеобразовательных школах. Он предоставляет достаточный набор операторов реализующих алгоритм ветвления, такие как case и сочетание операторов if else.

Для примера рассмотрим простейший алгоритм, который подскажет, что взять на улицу в зависимости от погоды. Изобразим алгоритм ветвления в виде псевдокода – некоторую систему обозначений и правил, который легко читается из-за схожести с естественным языком:

- 1) *Выйдем на балкон;*
- 2) *Посмотрим на небо;*
- 3) *Если есть облака, то возьмём зонт.*

В этом алгоритме фраза «если есть облака» является условием, а «то возьмем зонт» действием которое выполняется, если условие будет верным. Данный алгоритм является неполным алгоритмом, так как мы не указали другого действия на случай когда «нет облаков».

Добавим в наш алгоритм действие, если погода не предвещает дождь:

- 1) *Выйдем на балкон;*
- 2) *Посмотрим на небо;*
- 3) *Если есть облака, то возьмём зонт, иначе возьмём мяч.*

Алгоритм ветвления в данном примере становится полным, так как у нас появилось действие, выполняемое, в случае если условие «если есть облака» будет неверным, и соответственно действие «иначе возьмём мяч» выполнится, если погода безоблачная и пригодна для игры во дворе.

Когда алгоритм должен содержать больше вариантов развития событий используют конструкцию «переключатель» - это оператор выбора. Пример на псевдокоде будет выглядеть так:

1) *Вася получил оценку по информатике в дневник;*

2) *Выбор действия в зависимости от оценки:*

- если оценка 2, то строго побеседовать с сыном о его успеваемости;

- если оценка 3 или 4, то узнать каких знаний не хватило;

- если оценка 5, то похвалить;

- если нет оценки, то засомневаться был ли сын в школе.

Решение задачи с помощью алгоритма выбора уменьшает количество условий и сокращает код, что хорошо сказывается на его читаемости. Как и большинство современных высокоуровневых языков программирования, в языке Pascal написание операторов основывается на значении английских слов:

1) *if* — если;

2) *then* — то/тогда;

3) *else* — иначе;

4) *case* — случай/когда;

5) *of* — из.

В языке Pascal можно делать вложение условий, в другие условия, используя, специальные операторы скобок — *begin* и *end*. Применяв эти знания, опишем пример с погодой (листинг 1).

```

program vetvlenija1;
uses crt;
var NUM :integer;
begin
    write ('Введите 0 если нет облаков и 1 если они есть');
    readln (NUM);
    if NUM=1 then writeln ('Возьмём зонт')
    else writeln ('Возьмём мяч');
    readln();
end;

```

Листинг 1. Пример использование операторов if и else

Реализация алгоритма выбора действий родителя исходя из полученной в школе оценки (листинг 2).

```

program vetvlenija2;
uses crt;
var NUM :integer;
begin
    write ('Введите оценку из дневника Васи');
    readln (NUM);
    case NUM of
        2: writeln ('Серьёзно поговорить об успеваемости');
        3,4: writeln ('Помочь улучшить знания');
        5: writeln ('Похвалить');
    else
        writeln ('Узнать был ли Вася в школе');
    end;
end.

```

Листинг 2. Пример использование оператора case

Важно отметить несколько особенностей синтаксиса используемых команд. Если используется оператор `if` в конструкции неполного ветвления — то конец конструкции обозначается точкой запятой как обычно — после выполняемой команды, а если реализуется полное ветвление, то после первого оператора перед оператором `else` точка с запятой не ставятся, только после оператора, который расположен после оператора `else`, сообщая компилятору что конструкция закончена.

Оператор `case` обязательно должен иметь закрывающую скобку в виде оператора `end`; а также настоятельно рекомендуется обрабатывать значение, на которое не указан определенный выбор.

Это делается, если вдруг Васе из нашей программы поставили бы единицу или вообще не поставили бы оценку — варианта действий для такого значения особого нет, и в коде программы мы указали в конце перечисленных действий оператор `else` и действие, выполняемое в любых других случаях, если не сработал один из предыдущих. И последнее, если в операторе `case` у нас могут быть большое количество значений, а действия могут применяться в непрерывном рядах случаев, тогда удобнее записывать интервалами, используя конструкцию вида «2..4» вместо перечисления «2,3,4».

Для оператора `if` можно применять следующие операции сравнения:

- 1)  $x < u$  верно когда значение « $x$ » меньше значения « $u$ »;
- 2)  $x \leq u$  верно когда значение « $x$ » меньше или равно значения « $u$ »;
- 3)  $x = u$  верно когда значение « $x$ » равно значению « $u$ »;
- 4)  $x > u$  верно когда значение « $x$ » больше значения « $u$ »;
- 5)  $x \geq u$  верно когда значение « $x$ » больше или равно значению « $u$ »;
- 6)  $x \neq u$  верно когда значение « $x$ » не равно значению « $u$ ».

В языке Pascal можно комбинировать условия логическими операторами `and` (и), `or` (или), `not` (не).

Это удобно когда в алгоритме нужно сравнить несколько значение, например:

*if A<B and A>C then writeln('A меньше B и A больше C');*

## **1.2 Существующие ЦОРы для обучения ветвлению**

Из приказа № 345 от 28 декабря 2018 г. «О федеральном перечне учебников, рекомендуемых к использованию при реализации имеющих государственную аккредитацию образовательных программ начального общего, основного общего, среднего общего образования» [1], рекомендуемые к использованию учебники при реализации обязательной части основной образовательной программы предмета информатика (базовый уровень):

- 1) Босова Л. Л., Босова А. Ю. ООО «Бином. Лаборатория знаний» – Информатика. Базовый уровень. Учебник . 8 класс [3].
- 2) Поляков К. Ю., Еремин Е. А. ООО «Бином. Лаборатория знаний» – Информатика. Учебник 8 класс [7].
- 3) Семакин И.Г., Залогова Л.А., Русаков С.В., Шестакова Л. В. ООО «Бином. Лаборатория знаний» – Информатика. Учебник 9 класс [16].

Босова Л. Л., Босова А. Ю. для своего учебника предлагают для методической поддержки темы «§3.4. Программирование разветвляющихся алгоритмов» [4, с. 149] презентацию [17]. Презентация имеет хорошее изложение теоретического материала. Для работы с учащимися представлен 1 слайд с 15 вопросами, из которых 7 являются заданиями написать программу, остальные являются вопросами требующие устного ответа. Главный недостаток данного цифрового образовательного ресурса является необходимость индивидуальной проверки каждого учащегося. Также нет возможности быстро перейти к теоретическому материалу без закрытия показа слайдов и поиска нужного

слайда, если у учащихся возникнут вопросы во время написания программ. У презентации есть и технический недостаток – для классов с установленным свободным программным обеспечением, есть копия [19] в формате Open Document Format – и на некоторых слайдах замечен сдвиг текста, приводящий к затрудненному чтению.

Поляков К. Ю., Еремин Е. А. для своего учебника предлагают для методической поддержки темы «§19. Ветвления» [7, с.139], авторскую презентацию [8]. В ней представлен теоретический материал, хорошие и плохие примеры реализации алгоритмов ветвления. Стоит отметить универсальность презентации, так как примеры кода есть на языке программирования Python. Затронуты такие понятия как дерево решений, что расширяет понимание о применении алгоритмов ветвления. Материал подаётся последовательно в сопровождении вопросами к классу. Презентация не имеет интерактивных элементов, не считая анимацию отображения слайдов. К этому учебнику есть презентация, подготовленная М.Г. Кожиним, И.Е. Бондаренко, Е.А. Черноуцан и Н.Г. Неуйминой, учителями лицея № 110 им. Л.К. Гришиной (г. Екатеринбург) [5]. В ней представлен только теоретический материал.

Помимо презентаций к этому учебнику есть авторские тесты [9] для программы NetTest [11]. А.С. Башлаков предложил тесты [2] для программы MyTestXPro [15]. Также есть онлайн тестирование [10], содержащий 7 фиксированных задач. От учащихся требуется ввести значение переменной после выполнения приведенного в задаче кода.

К учебнику авторов Семакин И.Г., Залогова Л.А., Русаков С.В., Шестакова Л. В. Предлагается архив цифровыми образовательными ресурсами [17]. Однако для поддержки темы программирование ветвлений на Паскале [16] нет презентационного и проверочного материала. Для методической поддержки авторы рекомендуют использовать ЦОРы из единой коллекции цифровых образовательных ресурсов [13]. На сайте

единой коллекции цифровых образовательных ресурсов удалось найти презентацию [11] в которой кратко даётся только теоретический материал.

### **1.3 Тестирование в образовательном процессе**

Тестирование в педагогике выполняет три основные взаимосвязанные функции.

Диагностическая функция заключается в выявлении уровня знаний, умений, навыков учащегося. Это основная, и самая очевидная функция тестирования. По объективности, широте и скорости диагностирования, тестирование превосходит все остальные формы педагогического контроля.

Обучающая функция тестирования состоит в мотивировании учащегося к активизации работы по усвоению учебного материала. Для усиления обучающей функции тестирования, могут быть использованы дополнительные меры стимулирования, такие, как раздача преподавателем примерного перечня вопросов для самостоятельной подготовки, наличие в самом тесте наводящих вопросов и подсказок, совместный разбор результатов теста.

Воспитательная функция проявляется в периодичности и неизбежности тестового контроля. Это дисциплинирует, организует и направляет деятельность учащихся, помогает выявить и устранить пробелы в знаниях, формирует стремление развить свои способности [6].

## 1.4 Процесс обучения в виртуальной реальности

Одним из наиболее популярных направлений развития виртуальной и дополненной реальности является образование. Существует много различных вариантов применения современных технологий в этой области — от простых школьных туров по Древнему Египту на уроках географии до обучения специалистов для работы на сверхскоростном поезде или на космической станции [22].

Использование виртуальной реальности открывает много новых возможностей в обучении и образовании, которые слишком сложны, затратные по времени или дороги при традиционных подходах, если не всё одновременно. Можно выделить пять основных достоинств, применения AR/VR технологий в образовании.

**Наглядность.** Используя 3D-графику, можно детализировано показать химические процессы вплоть до атомного уровня. Причем ничто не запрещает углубиться еще дальше и показать, как внутри самого атома происходит деление ядра перед ядерным взрывом. Виртуальная реальность способна не только дать сведения о самом явлении, но и продемонстрировать его с любой степенью детализации.

**Безопасность.** Операция на сердце, управление сверхскоростным поездом, космическим шаттлом, техника безопасности при пожаре — можно погрузить зрителя в любое из этих обстоятельств без малейших угроз для жизни.

**Вовлечение.** Виртуальная реальность позволяет менять сценарии, влиять на ход эксперимента или решать математическую задачу в игровой и доступной для понимания форме. Во время виртуального урока можно увидеть мир прошлого глазами исторического персонажа, отправиться в путешествие по человеческому организму в микрокапсуле или выбрать верный курс на корабле Магеллана.

**Фокусировка.** Виртуальный мир, который окружает зрителя со всех сторон на все 360 градусов, позволяет целиком сосредоточиться на материале и не отвлекаться на внешние раздражители.

**Виртуальные уроки.** Вид от первого лица и ощущение своего присутствия в нарисованном мире — одна из главных особенностей виртуальной реальности. Это позволяет проводить уроки целиком в виртуальной реальности.

Любой из разработанных образовательных курсов может быть адаптирован для самостоятельного изучения. Сами уроки могут размещаться в онлайн-магазинах (например, Steam, Oculus Store, App Store, Google Play Market), чтобы у всех была возможность осваивать или повторять материал самостоятельно.

### **1.5 Выбор средств создания виртуальной реальности**

Для разработки цифрового образовательного ресурса использующего средства виртуальной реальности в рамках выпускной квалификационной работы используется игровой фреймворк. Это позволяет не углубляться в написание кода генерации виртуальной среды, и сокращает время разработки.

15 лет назад кодовая база для игр создавалась с нуля внутри компаний — разработчиков определенных продуктов. Немного позже (примерно в середине первого десятилетия XXI века) вслед за разработкой других компонентов игр, таких как арт или звук, создание движков вышло на аутсорсинг — выделилось в отдельную индустрию. Происходило это постепенно. Когда сформировалась обособленная кодовая индустрия, бизнес приобрел другой вид, появились движки широкой ценовой категории. В качестве примера можно привести Ogre, DarkGDK, HGE. Позже на рынок вышли другие, более прокачанные игроки, предложившие разработчикам полноценные движки: Torque 3D, Unity 3D, UDK.

Самый популярный движок для создания 2D- и 3D-игр в настоящее время является Unity3d [20]. Бесспорно, он стал лидером индустрии, и, как только появляется новая игровая/графическая технология, разработчики незамедлительно реализуют ее в Unity. Кроме разработки синглплеерных игр для PC, посредством подключаемых экспортеров можно портировать игры под другие ОС, консоли и мобильные технологии. Образовалась целая индустрия, работающая над созданием дополнений и расширений движка.

У редактора Unity есть порты под OS X и Windows, при этом изначально он был предназначен для OS X. В Unity включена поддержка DirectX 11, что открывает твоим приложениям дорогу в миры Windows 8 и Windows Phone 8. В версиях Unity выше 4.2, появилась поддержка последней на данный момент OpenGL ES 3.0, пока этими средствами обладают только самые современные Android-смартфоны. Движок Unity особенно ценен за низкий порог вхождения для начинающих программистов, благодаря этому, а также тому, базовая версия является бесплатной, вокруг движка организовалось огромное сообщество.

Низкий порог вхождения является результатом грамотного дизайна приложения: многие вещи можно выполнить с помощью различных редакторов, не написав при этом ни строчки кода (если что, код пишется на JavaScript, C#, Boo). Исходный код на C/C++ закрыт, но это в связи с расширенной компонентной структурой движка не создает никаких преград.

### **1.6 Реализации ЦОР на движке Unity3d**

Основой виртуального мира является набор из 4 сцен:

- 1) Основное поле для перехода между заданиями – сцена main (рисунок 1);
- 2) Сцена, формирующая виртуальный кабинет информатики – сцена classroom (рисунок 2);

- 3) Диалоговый режим представлен отдельной сценой – dialog (рисунок 3);
- 4) Финальный тест сделан отдельной сценой – test (рисунок 4).



Рис. 1. Сцена main виртуального мира



Рис. 2. Сцена classroom

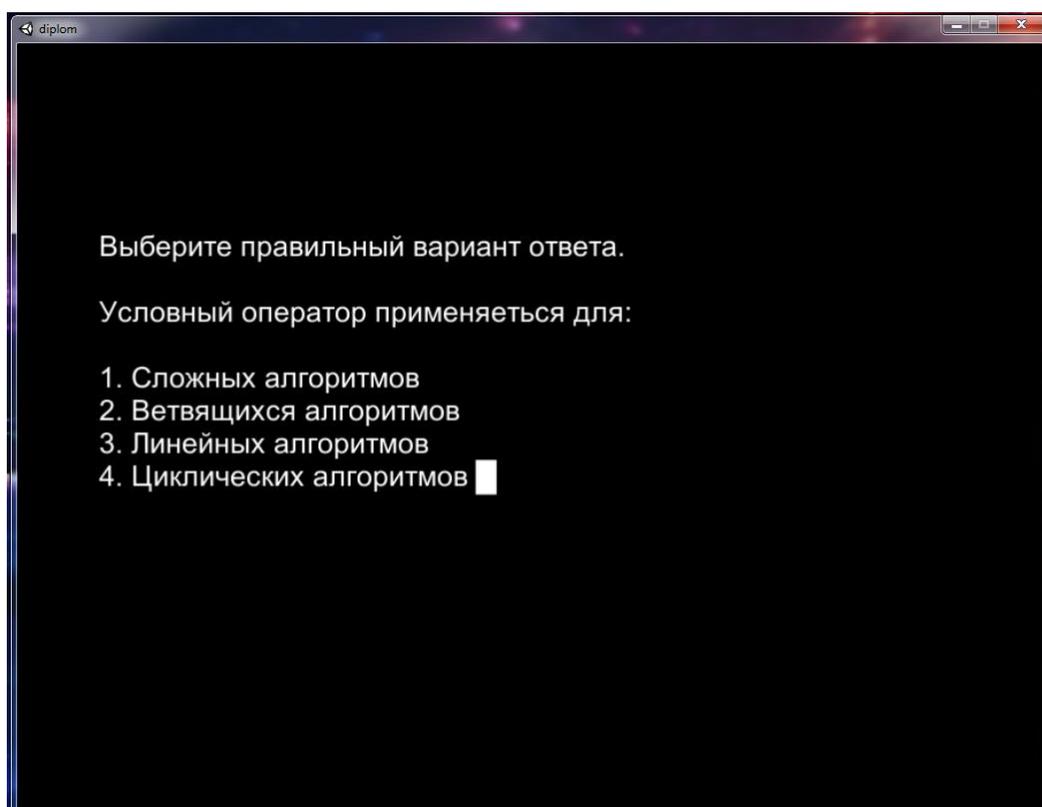


Рис. 3. Сцена dialog

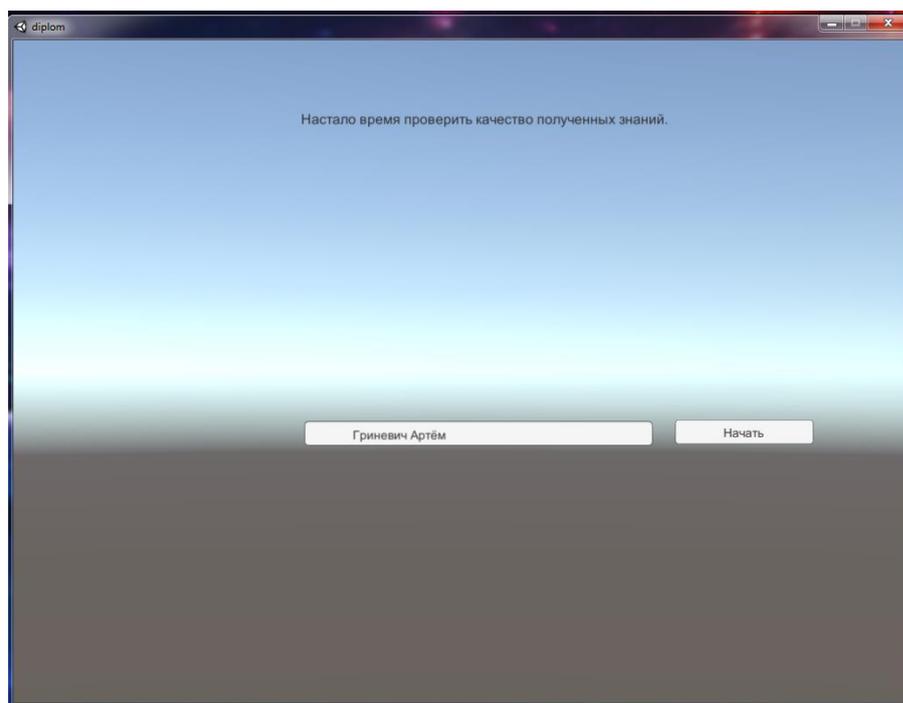


Рис. 4. Сцена test

Для переходов между сценами в виртуальном мире расположены 3d объекты с компонентами *colider*, соприкосновением с ними – происходят события *OnTriggerEnter*.

Сохранение состояния приложения, текущего уровня, подуровня реализовано применением паттерна программирования *singleton*. Экземпляр класса *StateMachina* реализованный в скрипте *StateMachina.cs*, при любой загруженной сцене активным является один и тот же экземпляр, в полях которого храниться состояние приложение, координаты возврата в мир *main*, уровень для выбора диалога отображения и другие свойства и методы.

Перед каждым виртуальным домом располагаются скрытые 3d объекты: *BeforeIfElseTrain*, *BeforeIfElseTheory*, *BeforeCaseTheory*, *BeforeTest*, *BeforeCaseTrain*, *BeforeFinalTrain* – к ним прикреплен компонент скрипт *SetReturnCoordinate.cs* – основная задача которого – записать координаты пользователя перед домом в который он вероятно войдет. Значение храниться у экземпляра *StateMachina* в свойстве *VectorReturnToScene* типа *Vector3* (трёхмерный вектор).

Для входа в сами виртуальные дома используются скрытые 3d объекты:

*EnterToIFElseTheory*, *EnterToCaseTheory*, *EnterToTest*, *EnterToCaseTrain*, *EnterToFinalTrain*, *EnterToIFElseTrain* – расположенные в дверных проемах виртуальных домов. К данным объектам добавлен компонент скрипт *EnterTheory1.cs*, который добавляет свойства для каждого экземпляра *level* – уровень. Для близлежащих объектов уровни располагаются по возрастанию. Также реализован метод *OnTriggerEnter* который сверяет уровень *StateMachina* и уровень самого объекта входа для блокирования преждевременного перехода к тестовым заданиям без прохождения теоретических диалогов. При последовательном исследованием учеником мира, уровень в данном случае будет возрастать с каждым переходом, в этом случае будут загружаться сцены *classroom*.

Сцена *classroom* выглядит как небольшой кабинет информатики. Активных элемента в нём два – входная дверь и ученический стол, стул. Для выхода из данной сцены на дверном проеме установлен прозрачный объект *ExitCollider* с компонентом скрипт *ReturnToMain.cs*, где на методе *OnTriggerEnter* реализована загрузка основной сцены *main*.

На объектах стол и стул добавлен компонент скрипт – *RunDialog.cs* в котором происходит на методе *OnTriggerEnter* происходит обнуление сохраненного под уровня, проверка текущего уровня и загрузка сцены *dialog* или *test* если уровень достиг 6.

В сцене *dialog* есть глобальный объект *StateMachina* и *DinamicText*. *StateMachina* содержит в себе динамический список диалогов разграниченных по уровням и подуровням, свойствам конечности уровня, полем с правильным ответом и признак диалога с вопросом. последовательно загружается нужный текст в соответствие уровня и под уровня. Последующий под уровень увеличивается на единицу, до тех пор пока загружаемый диалог не будет иметь признак конечности. Признак

конечности сообщает скрипту о том, что следующим загруженным этапом будет не следующий диалог, а увеличение текущего уровня и выход на сцену main. Если у диалога имеется признак вопроса – то от пользователя ожидается нажатие клавиш от 0 до 9 для выбора ответа, при правильном ответе возможен переход на следующий подуровень. Правильный и неправильный ответ выводится сразу после совершения выбора варианта ответа. При неверном ответе, загружается текст текущего диалога до тех пор, пока не будет выбран правильный ответ.

Сцена тестирования *test* имеет 3 объекта для взаимодействия с учащимся – тест, кнопка, поле ввода. Вопросы содержатся в динамическом списке объекта Questions. Редактирование доступно из среды unity editor. Выбор 10 заданий для теста осуществляется бесконечным циклом, случайному номеру от 0 до количества вопросов выбирается вопрос, и если он не выбран ранее для теста, присваивается порядковый номер до тех. Таким образом, реализован механизм выбора случайных 10 тестовых задания без повторений. Учащемуся после введение имени, последовательно предлагается ответить на 10 выбранных для него вопросов, после чего введенные учеником ответы сравниваются с известным ответом. При количестве верных ответов меньше 5 – выставляется оценка «2», от 5 до 7 – «3», от 7 до 9 – «4», если все 10 из 10 – то «5».

По завершению теста, кнопка ответить переименовывается в кнопку выход, на котором происходит завершение работы приложения.

Результат учащегося записывается в текстовый файл – rating.txt.

## Вывод по главе 1

Анализ цифровых образовательных ресурсов, предлагаемых к рекомендованным трем учебникам для школы на уроках информатики 8 и 9 классов, прослеживаются общие недостатки – проведение практической части урока, нет контроля учащихся. Выделился только учебный курс по учебнику Полякова К. Ю. – предоставлено несколько вариантов для тестирования учащихся, без возможности редактировать задания. В онлайн тесте – задания являются фиксированными и легко будут «списываться». Имеется возможность централизованного тестирования с применением программ «NetTest» [14] или «MyTestXPro» [15], однако данные программы специализируются только на контрольных мероприятиях, без возможности демонстрации теоретического материала или проведения практических работ по закреплению образовательного материала.

В худшую сторону выделился учебник авторов Семакина И.Г., Залоговой Л.А., Русакова С.В., Шестаковой Л. В. – предложив использовать общий коллекции образовательных ресурсов, на которых был найден только теоретический материал в формате презентации.

Наиболее комплексный подход к методической поддержке оказался у учебника авторов Босовой Л. Л., Босовой А. Ю. К учебнику сделаны по каждой теме своя презентация. В презентации по теме «Ветвления» предоставлен и теоретический материал, и вопросы для работы с классом, и контрольный материал. Однако формат цифрового образовательного ресурса в виде одной презентации не подразумевает взаимодействия учащихся с ресурсом. Погружением учащегося в виртуальную реальность, повышаем качество освоение знаний темы ветвления.

## **ГЛАВА 2. Методика обучения по теме «Ветвление на языке Pascal» с использованием средств виртуальной реальности**

### **2.1 Составление требований к цифровому обучающему ресурсу**

Примерная рабочая программа по предмету «*Информатика и ИКТ*» для 8 класса на 2018-2019 учебный год составляется на основе следующих нормативных документов:

- 1) Приказа Министерства Образования и Науки РФ от 17.12.2010 года №1897 «Об утверждении Федерального государственного образовательного стандарта основного общего образования» [12];
- 2) Примерной программы по учебному предмету «Информатика и ИКТ» авторской программы Босовой Л.Л., Босова А.Ю. Информатика. Программа для основной школы: 5-6 классы. 7-9 классы. – М.: БИНОМ. Лаборатория знаний, 2013 [4];
- 3) «Информатика: Учебник для 8 класса» авторов Босова Л.Л., Босова А.Ю. [3].

Программа для учащихся 8 класса и рассчитана на 35 часов (1 часа в неделю при шестидневной учебной неделе). На темы ветвления отводится 2 часа.

Тема первого подготовительного урока (1 час): Программирование разветвляющихся алгоритмов. Условный оператор. На уроке необходимо дать определения: *вещественный тип данных, целочисленный тип данных, символьный тип данных, строковый тип данных, логический тип данных*. Учащиеся должны уметь строить разветвляющийся линейный алгоритм, понимать составной оператор, строить алгоритмы в графическом виде, используя блок схемы, а также записывать его в виде псевдокода.

Тема второго урока (1 час): Многообразие способов записи ветвлений. На этом уроке непосредственно учащиеся знакомятся с синтаксисом условного оператора, сокращенной формой условного оператора, с записью составного оператора и реализации вложенных

ветвлений. Учащиеся после этого урока должны знать все способы записи ветвлений и уметь составлять программы, реализующие ветвление на языке Pascal.

Таким образом, проясняются требования к цифровому обучающему ресурсу:

- 1) ЦОР должен помочь в освоение реализации алгоритмов ветвления за 1 урок;
- 2) ЦОР должен иметь теоретический материал, содержащий все возможные формы записи условных операторов на языке Pascal;
- 3) Учащиеся, с помощью учителя используя ЦОР должны иметь возможность закрепить знания с применением коллективных действий;
- 4) В конце урока с помощью ЦОР учитель проверит качество знаний и умений учащихся, а также сможет выставить оценки в журнал полученные на основе результатов теста, который также должен быть включен в ЦОР;
- 5) У учителя должна быть возможность добавлять, удалять и редактировать контрольные вопросы, используемые на этапе тестирования учащихся.

## **2.2 Методика проведения урока «Ветвление»**

Для методической поддержки урока «Многообразие способов записи ветвлений», у учителя должна быть возможность демонстрации экрана всему классу средствами проектора и экрана, который должно быть хорошо видно. Разрешение экрана не менее 1024x768 пикселей.

Образовательная часть урока разбита на 3 части:

- 1) Изучение;
- 2) Закрепление;
- 3) Проверка знаний.

Проведение части закрепление и проверка знаний, требуется наличие компьютеров с операционной системой семейства windows у каждого учащегося.

## 2.3 Описание разработанного ЦОР «Ветвление»

При загрузке приложения, появляется диалоговое окно выбора качества изображения и разрешение монитора (рисунок 5).

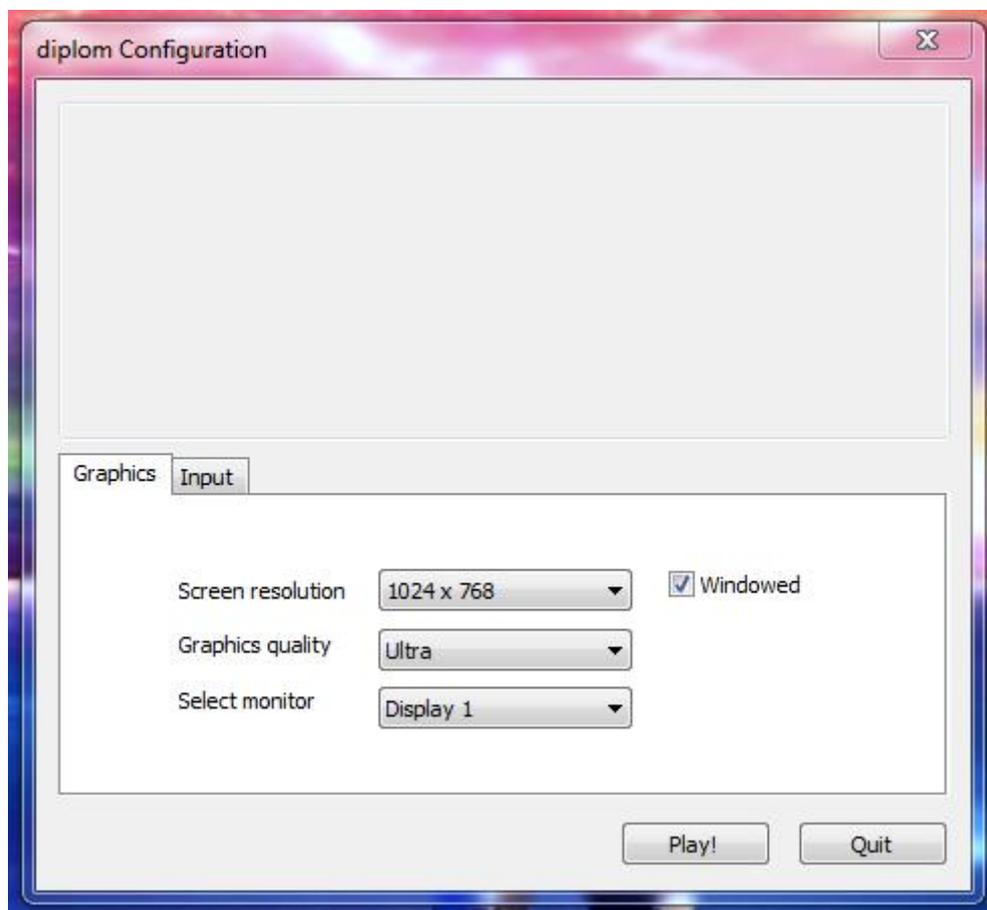


Рис. 5. Диалоговое окно выбора разрешения экрана

После выбора правильных настроек соответствующих установленным настройкам компьютера будет экран приветствия с описанием элементов управления (рисунок 6). Органами управления в виртуальном мире являются классические клавиши «W», «A», «S», «D» а также клавиши со стрелками. Для ориентирования рекомендуется использовать манипулятор мышь. В дальнейшем для прохождения тестов потребуется пользоваться цифровыми клавишами и клавишей пробела для перехода к следующим диалоговым окнам. Взаимодействие в виртуальном мире осуществляется путем касания с интересующим объектом.

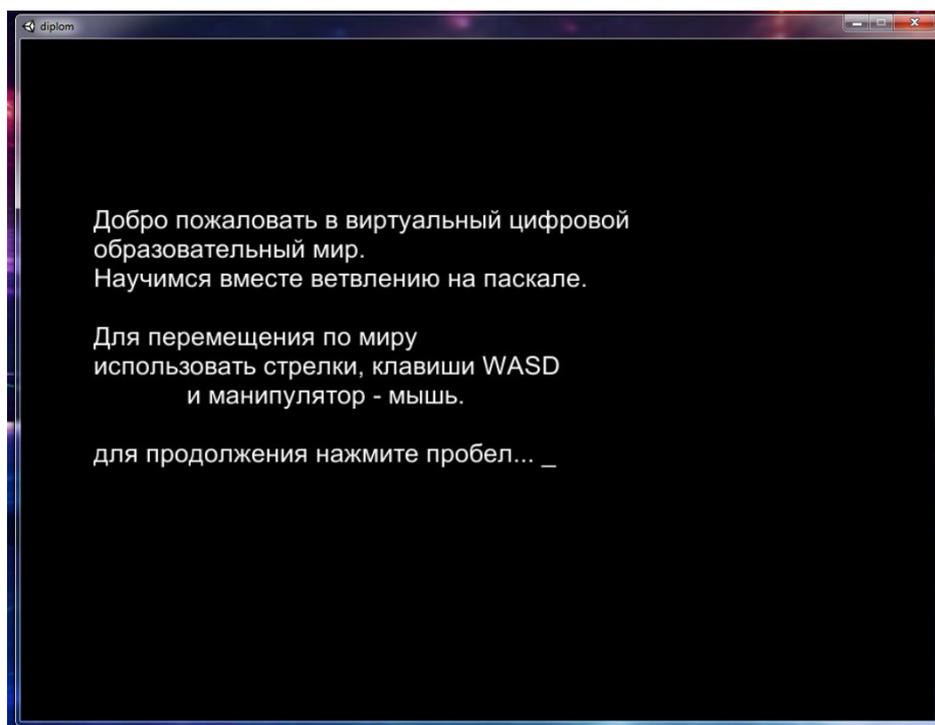


Рис.6. Экран приветствия

Нажатием клавиши пробел, цифровой образовательный ресурс загружает основную сцену виртуального мира (рисунок 7).



Рис. 7. Основная сцена виртуального мира

Учащемуся необходимо исследовать этот виртуальный мир и посетить несколько домов знаний (рисунок 8).



Рис. 8. Здание в виртуальном мире с теоретическими диалогами  
Внутри каждого здания загружается сцена classroom (рисунок 9), из которой можно выйти обратно на основную сцену или загрузить диалоги для данного уровня (рисунок 10).



Рис. 9. Вид сцены classroom

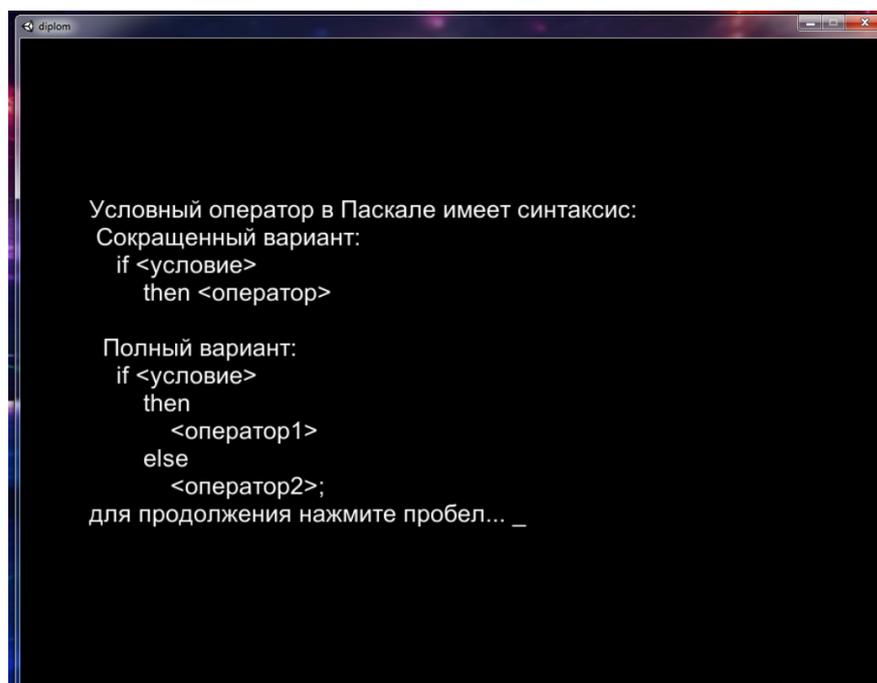


Рис. 10. Вид диалогового окна с теоретическим материалом

В процессе прохождения теоретического материала, пользователю становится доступным входить в другие виртуальные сооружения, например в дома тренировки (рисунок 11), которые дают учащемуся возможность закрепления знаний, выбрав, верный ответ для продолжения (рисунок 12).



Рис. 11. Внешний вид строений с закрепляющими диалогами

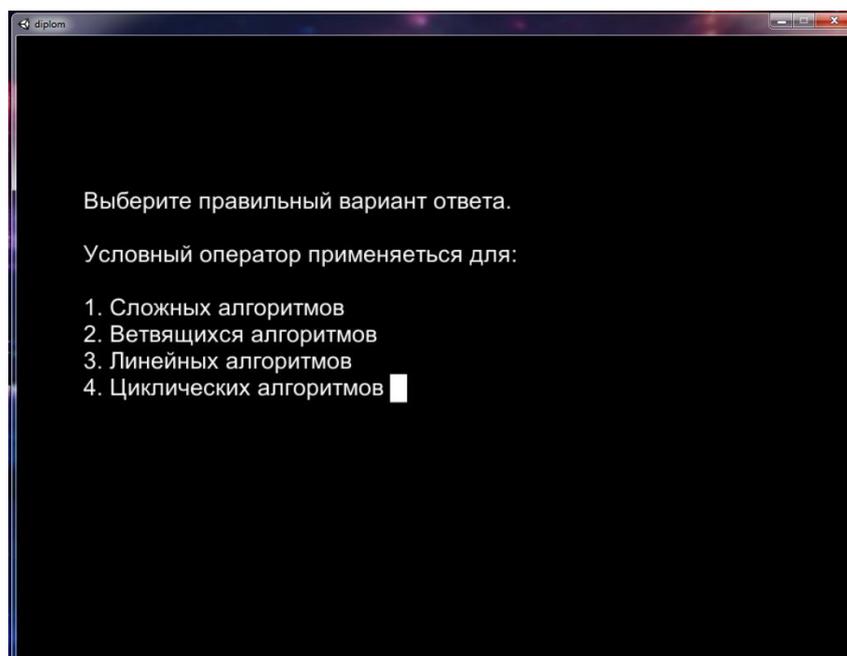


Рис. 12. Пример закрепляющего теста

При выборе правильного ответа, после нажатия пробела будет загружен следующий диалог (рисунок 13). При выборе неверного ответа – будет загружен повторно текущий диалог (рисунок 14).

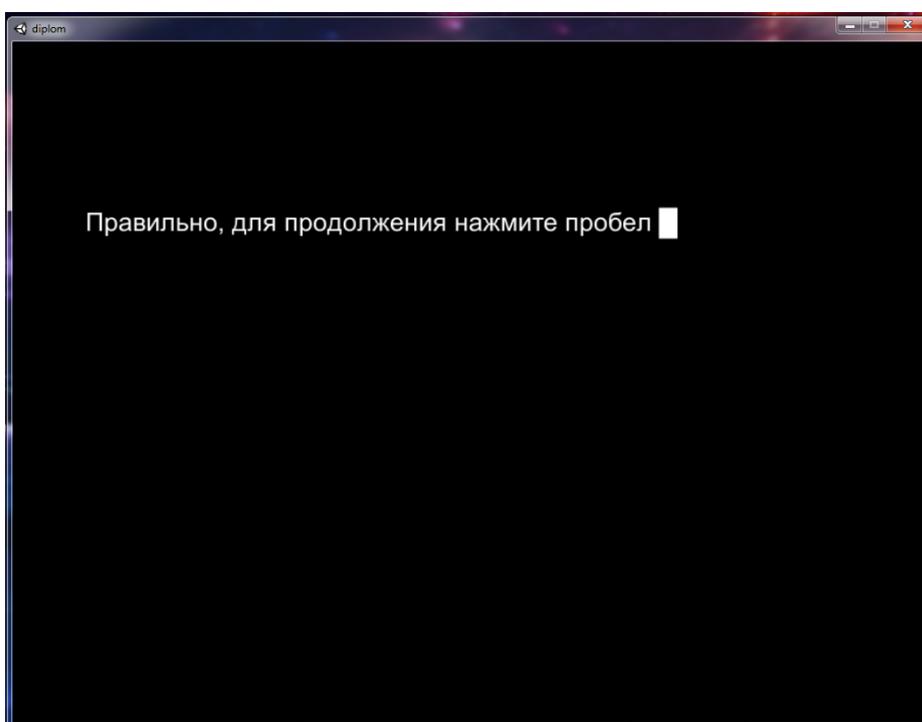


Рис. 13. Правильный ответ на тренировочный вопрос

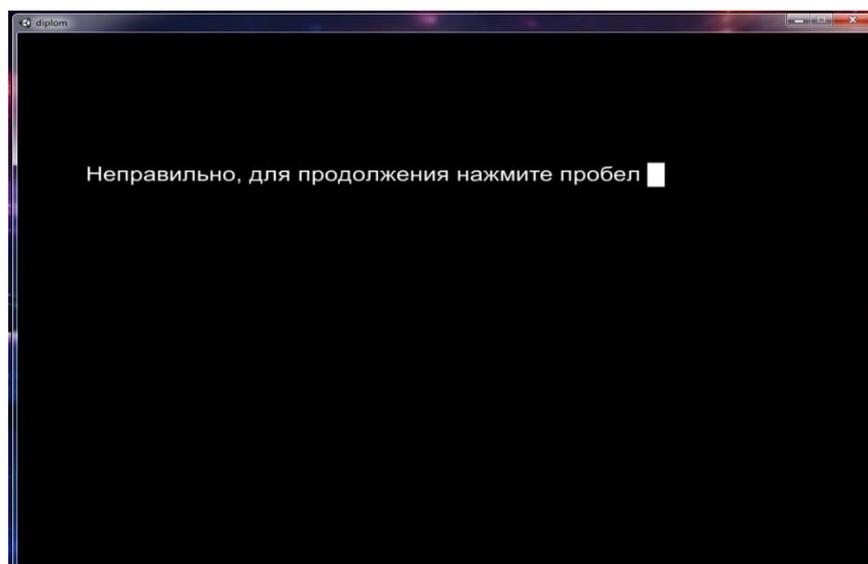


Рис. 14. Неправильный ответ на тренировочный вопрос

Исследовав теоретические виртуальные учебные классы, учащийся должен прийти до главного здания – проверки знаний (рисунок 15).



Рис. 15. Внешний вид здания проверки знаний

Если учащийся прошел весь тренировочный материал, то будет доступен сам тест. На первом экране теста, приложение попросит ввести имя и фамилию (рисунок 16). Нажатие кнопки «Начать» - загрузит первый вопрос из 10 вопросов выбранных программой.

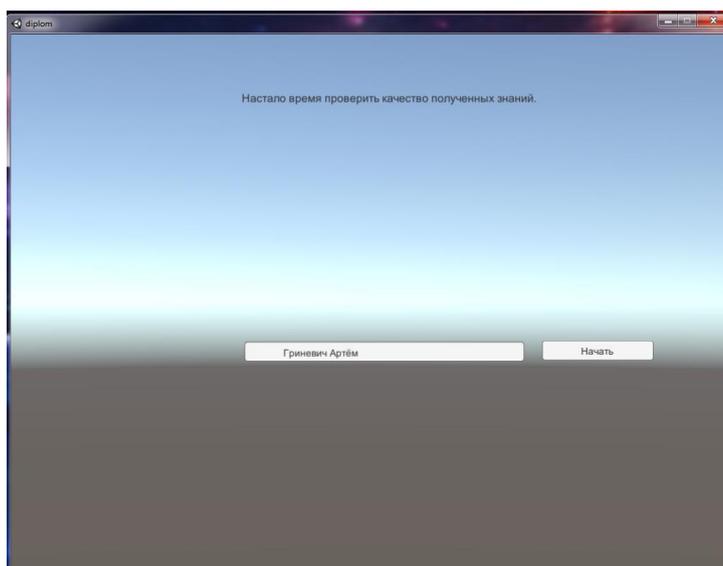


Рис. 16. Экран ввода имени учащегося

После ответа на каждое тестовое задание, будет подгружаться следующее (рисунок 17).

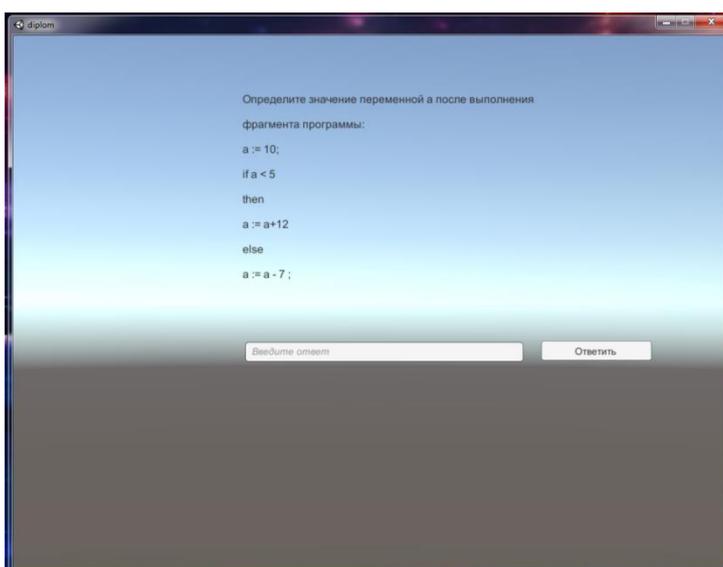


Рис. 17. Отображение тестового задания

## 2.4 Апробация результатов исследования

Педагогическая апробация проводилась в рамках научно-исследовательской практики в ГБПОУ "Южно-Уральский государственный колледж", города Челябинска. В рамках 1 половине пары рассмотрены возможности созданного цифрового образовательного ресурса.

Апробация прошла успешно. Способствовала этому правильная мотивация, цели и задачи изучения темы. Была отмечена качественная вовлечённость учащихся из-за использования средств виртуальной реальности.

Тема курса оказалась знакома для учащихся, и они быстро включились в работу. Современные дети особенно заинтересованы в получении знаний, особенно если информация представляется в виртуальном мире.

## **Вывод по главе 2**

В ходе практической части, был проведен анализ нормативных документов. Рассмотрены возможности создания приложений виртуальной реальности в образовательных целях. Составлен урок на тему ветвления. Обозначены цели и задачи, а также описаны знания и умения, полученные после прохождения этого курса.

Разработан ЦОР комплексного вида, позволяющий провести полностью урок по теме «ветвления» в курсе программирования на языке Pascal.

Реализованы этапы обучения, закрепления, и проверки знаний. В результате прохождения контрольного теста, учителю предоставляется результат проверки знаний в виде скрытого текстового файла, который можно использовать в качестве оценки учащегося для классного журнала.

## Заключение

В ходе исследовательской работы была рассмотрена теоретическая часть урока информатики по теме «Многообразие способов записи ветвлений». Современным задачам, решаемым с помощью программных продуктов, нужны понимание, как работает ветвление, какие способы задания условий существуют, и как реализуется алгоритм выбора.

Были найдены существующие цифровые образовательные ресурсы, отмечены их достоинства и недостатки. В практической части разработан цифровой обучающий ресурс с комплексным подходом к методической поддержке урока информатики по теме ветвление. Описана методика обучения с применением этого инновационного образовательного ресурса, включающая основные этапы образовательной деятельности – обучение, закрепление и контроль знаний.

Разработан цифровой обучающий ресурс с элементами средств виртуальной реальности, для проведения комплексного урока по теме ветвление на языке Pascal. Данная разработка включает в себя небольшой виртуальный мир, а описанные алгоритмы реализуют механизмы обучения, закрепления и тестирования учащихся. При этом код использует алгоритмы ветвления практически в каждой конструкции. Таким образом, было подтверждено необходимость знаний об условных операторах и умение составлять программы, даже для написания обучающего ресурса. В виду комплексного подхода и экономии времени на переключение, получившийся образовательный ресурс повышает качество освоения темы ветвления на уроках предмета информатика.

## Список литературы

1. Банк документов министерства образования Российской Федерации. [Электронный ресурс]. Приказ о федеральном перечне учебников №345 – Режим доступа: <https://docs.edu.gov.ru/document/1a542c2a47065cfbd1ae8449adac2e77/>, свободный.
2. Башлаков, А.С. Тесты к учебнику Поляков К. Ю. Информатика. Учебник 8 класс. Для программы MyTestXPro [Электронный ресурс]. – Режим доступа: <https://yadi.sk/d/sF4JISf13NofGd>, свободный
3. Босова, Л. Л. Информатика. 9 класс [Текст]: учебник / Л. Л. Босова, А. Ю. Босова. – 6-е изд. – М.: БИНОМ. Лаборатория знаний, 2016. – 160 с.: ил.
4. Босова, Л.Л. Методическое пособие. Информатика 7-9 классы [Электронный ресурс]. – Режим доступа: <http://lbz.ru/metodist/iumk/informatics/files/bosova-7-9-met.pdf>, свободный.
5. Кожин, М.Г. презентация к учебнику Поляков К. Ю. Информатика. Учебник 8 класс. [Электронный ресурс]. – Режим доступа: <https://yadi.sk/d/vdMvE6kw3K8T2Q>, свободный.
6. Магранова, Ю.В. Теория тестирования как основа оценивания уровня знаний в современной системе образования. [Текст]: Методы социологических исследований: сб. статей Магранова Ю.В. – М.:Теис, 2006. -С. 199-208.
7. Поляков, К. Ю. Информатика. Учебник 8 класс [Текст]: учебник / К. Ю. Поляков, Е. А. Еремин М.: БИНОМ. Лаборатория знаний, 2018. - 247с.:ил.
8. Поляков, К. Ю. Авторская презентация к учебнику Информатика. Учебник 8 класс. [Электронный ресурс]. – Режим доступа: <http://kpolyakov.spb.ru/download/slides8-3p.zip>, свободный.

9. Поляков, К. Ю. Тесты к учебнику Информатика. Учебник 8 класс. Для программы NetTest [Электронный ресурс]. – Режим доступа: <http://kpolyakov.spb.ru/loadstat.php?f=/download/tests8a.zip>, свободный.

10. Поляков, К. Ю. Онлайн тесты к учебнику Информатика. Учебник 8 класс. [Электронный ресурс]. – Режим доступа: <http://kpolyakov.spb.ru/school/test8a/p21.htm>, свободный.

11. Презентация на тему ветвления на языке Pascal [Электронный ресурс]. – Режим доступа: <http://school-collection.edu.ru/catalog/res/76cae59c-edcf-4008-a7c3-c1272e02292c/view/>, свободный.

12. Приказа Министерства Образования и Науки РФ от 17.12.2010 года №1897 «Об утверждении Федерального государственного образовательного стандарта основного общего образования» [Электронный ресурс]. – Режим доступа: <https://mosmetod.ru/metodicheskoe-prostranstvo/documenti/prikaz-ot-17-dekabrya-2010-g-1897.html>, свободный.

13. Сайт единой коллекции цифровых образовательных ресурсов [Электронный ресурс]. – Режим доступа: <http://school-collection.edu.ru>, свободный.

14. Сайт программы для тестирования NetTest [Электронный ресурс]. – Режим доступа: <http://kpolyakov.spb.ru/prog/nettest/nettnew.htm>, свободный.

15. Сайт программы для тестирования MyTestXPro [Электронный ресурс]. – Режим доступа: <http://mytest.klyaksa.net/htm/download/index.htm>, свободный.

16. Семакин, И.Г. Информатика и ИКТ [Текст]: учебник для 9 класса/ И.Г. Семакин, Л.А. Залогова, С.В. Русаков, Л. В. Шестакова М.: БИНОМ. Лаборатория знаний, 2016. – 208с.:ил.

17. Цифровые образовательные ресурсы к учебнику Семакина И.Г., Залогова Л.Ф., Информатика и ИКТ для 7-9 класса [Электронный ресурс]. – Режим доступа: <http://lbz.ru/files/6427/>, свободный.

18. Электронная презентация к учебнику Босовой Л.Л., Босовой А.Ю., Информатика, 8 класс. [Электронный ресурс]. – Режим доступа: <http://lbz.ru/metodist/authors/informatika/3/files/eor8/presentations/8-3-4.ppt> , свободный.

19. Электронная презентация к учебнику Босовой Л.Л., Босовой А.Ю. Информатика, 8 класс. [Электронный ресурс]. – Режим доступа: <http://files.lbz.ru/authors/informatika/3/8-3-4.odp> , свободный.

20. Unity для всех [Электронный ресурс]. – Режим доступа: <https://unity3d.com/ru/get-unity/download> , свободный.

21. Unity3D по русски. Русскоязычное сообщество Unity3d.ru [Электронный ресурс]. – Режим доступа: <http://www.unity3d.ru/>, свободный.

22. VRGeek, сайт о виртуальной реальности [Электронный ресурс]. – Режим доступа: <https://vrgeek.ru/>, свободный.

## Приложение 1

### Технологическая карта занятия по теме «Ветвление»

**Предмет:** Информатика

**Тема урока:** Многообразие способов записи алгоритмов ветвления

**Тип урока:** проверка знаний и объяснение нового материала.

#### Прогнозируемые результаты:

##### личностные:

- учить проверять себя;
- учить давать оценку своим действиям.

##### метапредметные:

- учить анализировать и выделять общее;
- учить находить наиболее оптимальный алгоритм действий.

##### предметные:

- вспомнить понятие алгоритма;
- определить случаи применения алгоритма ветвления;
- учить синтаксису операторов ветвления на языке Pascal;

**Дидактические средства:** учебник, памятки, карточки с заданиями для групповой работы.

**Оборудование:** Компьютерный класс, индивидуальные компьютеры для каждого учащегося, с установленной операционной системой Windows

Таблица 1.

## Структура урока по теме «Ветвление»

Этап урока	Виды работы, формы, методы, приемы	Содержание педагогического взаимодействия		Формируемые УУД	Планируемые результаты
		Деятельность учителя	Деятельность обучающихся		
Мотивация к учебной деятельности	Словесное приветствие .	Приветствует детей, проверяет их готовность к уроку. Настраивает на активную работу.  - Прозвенел уже звонок.	Организовывают рабочее место. Здороваются с учителем.	Личностные: управление своим настроением, умение выразить эмоции.  Метапредметные: организовывать рабочее место, настраиваться на	Организовать детей.  Проверить готовность к уроку.

		<p>Начинаем наш урок. Добрый день!</p> <p>- Вы готовы к нашему уроку?</p>		<p>познавательную деятельность.</p>	
	<p>Интерактивная игра «Поделись позитивом».</p>	<p>- С каким настроением вы пришли на урок?</p> <p>- Те, у кого хорошее настроение, пусть встанут.</p> <p>- Подойдите к своим товарищам, которые сегодня «не в духе» и</p>	<p>Отвечают на вопросы, обмениваются позитивными репликами.</p>		<p>Создать позитивное настроение.</p>

		<p>передайте им частичку своего позитива: скажите приятные слова, улыбнитесь и пожмите руку, обнимитесь и т.д.</p>			
<p>Актуализация знаний</p>	<p>Вспомним что такое «алгоритмы». Когда необходимо ветвление. (фронтальн</p>	<p>Проводит беседу о алгоритмах, как усложнять алгоритмы для решение практических задач. Приводиться пример с погодой: “На небе тучи,</p>	<p>Отвечают на вопросы.</p>	<p>Личностные: проявлять старательность, восстанавливать знания полученные ранее Предметные: Построение задачи на реализацию</p>	<p>Воспитывать внимательность, формировать навыки построения алгоритмов ветвления из задач.</p>

	о)	ВЗЯТЬ ЗОНТ»		ветвлений в программирование	
Постановка учебной задачи	Инструктаж по запуску цифрового образовательного ресурса (фронтально) о)	Учитель записывает на доске название программы для запуска виртуальной обучающей среды	Выполняют загрузку операционной системы на своих компьютерах.	Личностные: проводят самооценивание, определяют уровень своих знаний, формируется стремление к познанию. Метапредметные: учатся анализировать, сопоставлять, делать выводы,	Создать проблемную ситуацию, подтолкнуть детей к осознанию необходимости в получении новых знаний. Получить оценку работы учащихся на основании качества усвоенных знаний.

				<p>ставить цели познавательной деятельности.</p> <p>Предметные:</p> <p>Изучают способы записи ветвлений на языке паскаль,</p>	
	<p>Формулирование темы и целей урока (фронтально)</p>	<p>Проводит параллель с ранее изученным материалом.</p> <p>Организует фиксацию интеллектуальных затруднений при выполнении задания.</p>	<p>Начинают проходить задания в виртуальном мире.</p>	<p>проходят предварительные закрепляющие тесты, проходят контрольный тест.</p>	

		Инициализирует заинтересованност ь к прохождению заданий в виртуальной среде.			
--	--	---	--	--	--

## **Подробный конспект занятия**

Тип урока: проверка знаний и объяснение нового материала.  
Самообучение.

Цели урока: дать представление о различных способах записи ветвления на языке Pascal.

Методы обучения: беседа, пример, проблема, обсуждение, погружение в виртуальный образовательный мир.

Средства: персональный компьютер ПО, цифровой образовательный ресурс «Ветвление»

### **План урока:**

1. Организационный момент (1 мин).
2. Проверка домашнего задания (7 мин).
3. Актуализация. (5 мин).
4. Объяснение нового материала (15 мин).
5. Закрепление (12 мин)
6. Домашнее задание (2 мин).
7. Итоги урока (3 мин).

### **Ход урока**

#### **I. Организационный момент.**

- Поздороваться с классом.
- Отметить отсутствующих.
- Проверить готовность класса к уроку.

#### **II. Проверка домашнего задания.**

У вас было домашнее задание: повторить основные команды редактирования и форматирования ТП Word. Давайте проверим, как вы освоили эту тему, насколько хорошо вы запомнили последовательности команд для редактирования или форматирования фрагментов текста.

#### **III. Актуализация.**

Вопросы для обсуждения и поддержания беседы.

Что такое алгоритмы?

Какие числа алгоритмы вы знаете?

А внутри компьютера, какая система счисления используется?

#### **IV. Объяснение нового материала.**

Загружаем компьютеры, и запускаем приложение «ветвления». Проходим вместе коллективно часть обучения и закрепления внутри виртуального мира.

Исследуем мир, находим подсказки и следуем к следующему этапу.

#### **V. Закрепление.**

В этапах закрепления ученики, отвечают на вопросы предоставленные программой:

1. Условный оператор применяется для программирования:
  - Сложных алгоритмов
  - Ветвящихся алгоритмов
  - Линейных алгоритмов
  - Циклических алгоритмов
2. Ветвящийся алгоритм подразумевает:
  - Неоднократное повторение отдельных частей программы
  - Выполнению лишь нескольких, удовлетворяющих условию частей программы
  - Последовательное выполнении всех элементов
  - Верного ответа нет
3. Верно ли записан оператор выбора IF <оператор1> THEN <оператор2> ELSE <оператор3>?
4. Возможно ли неполное написание условного оператора?
5. Оператор CASE служит для создания:
  - Сложных алгоритмов
  - Ветвящихся алгоритмов
  - Линейных алгоритмов

- Циклических алгоритмов

6. Верно ли записан оператор CASE?

Case K of:

A1: <оператор 1>;

A2: <оператор 2>;

...

An: <оператор n>;

Else <оператор n+1>;

End;

7. Какое будет значение переменной K после выполнения фрагмента программы

a:=7;

b:=8;

k:=5+a;

if a>b then k:=k\*3

else k:=k-3;

writeln (k);

Продолжите фразу. Алгоритм включает в себя ветвление, если ...

- он составлен так, что его выполнение предполагает многократное повторение одних и тех же действий
- ход его выполнения зависит от истинности тех или иных условий
- его команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий

#### **VI. Домашнее задание.**

(Объяснить суть домашнего задания.)

#### **VII. Итоги урока.**

Что нового узнали для себя на уроке, и что вам уже было знакомо?

Каково ваше представление о числах сейчас, когда вы узнали о существовании других СС? Какие моменты вам были не понятны?

## **Анализ занятия.**

### **I. Цели**

Сформировать представление о ветвлении как одной из основных алгоритмических конструкций.

Изучить команду ветвления: назначение, формат записи.

Научить использовать конструкцию ветвления для решения задач.

### **II. Место в курсе (в разделе «Алгоритмизация»)**

Варианты:

1. Следование => Ветвление => Цикл.

2. Цикл с параметром (повтор n раз) => Ветвление => Цикл с условием.

3. Следование => Вспомогательные алгоритмы (подпрограммы) => Ветвление.

**III. Мотивация:** предложить задачу, которую нельзя решить без ветвления, интерес к виртуальному миру.

### **IV. Содержание теоретической части**

Определение ветвления. Описание его назначения (выбор одного из двух способов действий в зависимости от выполнения или невыполнения некоторого условия). Оператор выбора.

Форма записи: Если

Полное и сокращенное ветвление.

Многократное вложенное ветвление (выбор не из двух, а из нескольких действий).

**ЗАПИСЬ УСЛОВИЙ:** логические операции, логические выражения (например, сравнение двух величин). Составное условие (например, если надо сравнить 3 величины).

### **V. Содержание практической части (самостоятельно)**

Решение задач. Обращение к справочной информации.

## **VI. Контроль**

Тестирование с помощью ЦОР.

## Приложение 2

### Вопросы для теста учащегося

Определите значение переменной  $a$  после выполнения фрагмента программы:

```
a := 10;  
if a < 5  
  then a := 5;
```

Определите значение переменной  $b$  после выполнения фрагмента программы:

```
b := 1;  
if b >= 1  
  then b := b+1;
```

Определите значение переменной  $a$  после выполнения фрагмента программы:

```
a := 10;  
if a > 5  
  then a := 12;
```

Определите значение переменной  $a$  после выполнения фрагмента программы:

```
a := 10;  
if a < 5  
  then a := a+12  
  else a := a - 7 ;
```

Определите значение переменной  $b$  после выполнения фрагмента программы:

```
b := 10;  
if b < 15  
  then b := b+5  
  else b := b - 7 ;
```

Определите значение переменной a после выполнения фрагмента программы:

```
a :=4;
b := 1;
if (a > b) and (b > 2)
    then a := a-1;
```

Определите значение переменной a после выполнения фрагмента программы:

```
a :=4;
b := 1;
if (a < b) or (b < 2)
    then a := a-1;
```

Определите значение переменной a после выполнения фрагмента программы:

```
a :=7;
if not (a<0)
    then a := a-7;
```

Определите значение переменной c после выполнения фрагмента программы:

```
a :=1;
b :=3;
c: = -1;
if (a>0) and (b<10)
    then c := c+a-b;
```

Определите значение переменной c после выполнения фрагмента программы:

```
a :=1; b :=3; c: = -1;
if (a>0) or (b>10) then c := c+a-b;
```

Определите значение переменной c после выполнения фрагмента программы:

```
a :=1; b :=3; c: = -1;  
if c>-1 then c := a+b;
```

Определить что выведет программа после выполнения фрагмента программы:

```
I:=7;  
case I of  
0..1: Writeln(7);  
2,9: Writeln(0);  
Else Writeln(-1);  
end;
```

Определить что выведет программа после выполнения фрагмента программы:

```
I:=2;  
case i of  
0..1: Writeln(7);  
2,9: Writeln(0);  
Else Writeln(-1);  
end;
```

Определить что выведет программа после выполнения фрагмента программы:

```
I:=7;  
case i of  
0..1: Writeln(0);  
2..9: Writeln(7);  
Else Writeln(-1);  
end;
```

Определить что выведет программа после выполнения фрагмента программы:

```
I:=2;  
  case i of  
0..2: Writeln(7);  
3,5,9: Writeln(-1);  
Else Writeln(0);  
end;
```

Определить что выведет программа после выполнения фрагмента программы:

```
I:=5;  
  case i of  
0..2: Writeln(1);  
3,5,9: Writeln(2);  
Else Writeln(0);  
end;
```