**Муниципальное казенное образовательное учреждение**

**средняя общеобразовательная школа № 6 с.Гофицкое**

|  |  |  |
| --- | --- | --- |
| Рассмотрено на заседании методического объединения Протокол №\_\_\_\_\_\_от \_\_.\_\_\_.2014 г.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Согласовано  Зам.директора по УВР  \_\_.\_\_\_\_\_\_\_\_\_\_\_.2014г.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Т.П.Калмыкова |  | Утверждено  Директор МКОУ СОШ №6  от \_\_.\_\_.2014 года  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д.Ю.Васянкин  М.П. |

**Программа элективного курса по информатике**

**«ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ПРИМЕРЕ VISUAL BASIC»**

|  |  |
| --- | --- |
| Разработчик: | Болотова С.А.,  учитель информатики и ИКТ |
| квалификационная категория: | соответствие занимаемой должности |

c.Гофицкое, 2014г.

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

Курс «Основы программирования на примере Visual Basic» составлен на основе методического пособия М.Н.Бородина «Информатика. Программы для общеобразовательных учреждений 2-11 классы: методическое пособие»– М.: БИНОМ. Лаборатория знаний, 2010.

Курс ориентирован на учащихся 10-11 классов общеобразовательной школы, желающих расширить свои представления в области информатике.

**Цель курса** – обучение основам объектно-ориентированного программирования с использованием системы программирования Visual Basic, знакомство с конструированием объектов, подробно рассматривая каждый этап конструирования на большом количестве примеров. Основное внимание уделяется этапу проектирования задач и разработке информационной модели изучаемого объекта или системы.

**Задачи курса:**

* привить учащимся навыки, требуемые большинством видов современной деятельности (налаживание контактов другими членами коллектива, планирование и организация совместной деятельности и т. д.)
* сформировать умения решения исследовательских задач;
* сформировать умения решения практических задач, требующих получения законченного продукта;
* развить способность к самообучению.

Курс рассчитан на учеников, имеющих базовую подготовку по информатике. В основе курса лежит формирование теоретической базы и овладение учащимися конкретными навыками использования компьютерных технологий в сфере деятельности – программировании.

К теоретической базе относится знание общих принципов решения задач с помощью компьютера, понимание того, что значит поставить задачу и построить компьютерную модель, знание основных способов алгоритмизации. Навыки использования информационных технологий предполагают умения работать с готовыми программными средствами.

В соответствии с этим занятия делятся на теоретическую и практическую части. На теоретической части создаются компьютерные модели и алгоритмы решения задач. В ходе практических работ учащиеся пишут программы и проводят компьютерные эксперименты.

**Компьютерный практикум.** Разработка каждого проекта реализуется в форме выполнения практической работы на компьютере (компьютерный практикум). В учебном пособии содержатся подробные указания по построению компьютерных моделей и их реализации в форме проектов на языках программирования и в электронных таблицах.

Кроме разработки проектов под руководством учителя учащимся предлагаются практические задания для самостоятельного выполнения. В учебном пособии содержатся указания по их выполнению, а на Microsoft-CD хранятся готовые проекты на языке объектно-ориентированного программирования Visual Basic.

**Место предмета в федеральном базисном учебном плане**

Согласно федеральному базисному учебному плану для образовательных учреждений Российской Федерации на изучение курса отводится по 1 часу в неделю в течение двух лет обучения - 10-11 классы; 10 класс - 35 часов, 11 класс – 35 часов, всего 70 учебных часов.

**Методы преподавания и изучения:**

В основу работы с учащимися по изучению курса «Основы программирования на примере Visual Basic» положена методика, базирующаяся на следующих принципах развивающего обучения:

1. принцип обучения на высоком уровне трудности;
2. принцип ведущей роли теоретических знаний;
3. принцип концентрированности организации учебного процесса и учебного материала;
4. принцип группового или коллективного взаимодействия;
5. принцип полифункциональности учебных заданий.

Основным методом обучения в данном курсе является метод проектов. Проектная деятельность позволяет развить исследовательские и творческие способности- учащихся. В процессе обучения учитель кратко объясняет новый материал и ставит задачи, а затем консультирует учащихся в процессе решения этих задач. Учащиеся решают задачи, в основном практические, реализуя проекты по созданию приложений на компьютере (компьютерный практикум). Подробные указания по построению компьютерных моделей и их реализации в форме проектов на языке программирования Visual Basic содержатся в учебном пособии к курсу.

**Требования к уровню подготовки школьников**

В результате изучения курса учащиеся **должны**:

* Объяснять структуру основных алгоритмических конструкций и уметь их применять для построения алгоритмов;
* Знать основные типы данных и процедуры для Visual Basic;
* Знать этапы проектирования объектов;
* Уметь создавать проекты с использованием среды визуального объектно-ориентированного программирования Visual Basic;
* Уметь составлять и проводить поэтапное проектирование, осуществлять компьютерный эксперимент.

## Метапредметные образовательные результаты

Основные ***метапредметные образовательные результаты***, достигаемые в процессе изучения курса школьниками в области информатики и математики:

* уверенная ориентация учащихся в различных предметных областях за счет осознанного использования при изучении школьных дисциплин таких общепредметных понятий как «объект», «система», «модель», «алгоритм», «исполнитель» и др.;
* владение основными общеучебными умениями информационно-логического характера: анализ объектов и ситуаций; синтез как составление целого из частей и самостоятельное достраивание недостающих компонентов; выбор оснований и критериев для сравнения, сериации, классификации объектов; обобщение и сравнение данных; подведение под понятие, выведение следствий; установление причинно-следственных связей; построение логических цепочек рассуждений и т.д.,
* владение умениями организации собственной учебной деятельности, включающими: целеполагание как постановку учебной задачи на основе соотнесения того, что уже известно, и того, что требуется установить; планирование – определение последовательности промежуточных целей с учетом конечного результата, разбиение задачи на подзадачи, разработка последовательности и структуры действий, необходимых для достижения цели при помощи фиксированного набора средств; прогнозирование – предвосхищение результата; контроль – интерпретация полученного результата, его соотнесение с имеющимися данными с целью установления соответствия или несоответствия (обнаружения ошибки); коррекция – внесение необходимых дополнений и корректив в план действий в случае обнаружения ошибки; оценка – осознание учащимся того, насколько качественно им решена учебно-познавательная задача;
* владение основными универсальными умениями информационного характера: постановка и формулирование проблемы; поиск и выделение необходимой информации, применение методов информационного поиска; структурирование и визуализация информации; выбор наиболее эффективных способов решения задач в зависимости от конкретных условий; самостоятельное создание алгоритмов деятельности при решении проблем творческого и поискового характера;
* владение базовыми навыками исследовательской деятельности, проведения виртуальных экспериментов; владение способами и методами освоения новых инструментальных средств;
* владение основами продуктивного взаимодействия и сотрудничества со сверстниками и взрослыми: умение правильно, четко и однозначно сформулировать мысль в понятной собеседнику форме; умение осуществлять в коллективе совместную информационную деятельность, в частности при выполнении проекта; умение выступать перед аудиторией, представляя ей результаты своей работы с помощью средств ИКТ; использование коммуникационных технологий в учебной деятельности и повседневной жизни.

## Личностные образовательные результаты

* широкие познавательные интересы, инициатива и любознательность, мотивы познания и творчества; готовность и способность учащихся к саморазвитию и реализации творческого потенциала в духовной и предметно-продуктивной деятельности за счет развития их образного, алгоритмического и логического мышления;
* готовность к повышению своего образовательного уровня и продолжению обучения с использованием средств и методов информатики и математики;
* интерес к математики, стремление использовать полученные знания в процессе обучения другим предметам и в жизни;
* способность увязать учебное содержание с собственным жизненным опытом и личными смыслами, понять значимость подготовки в области математики и информатики в условиях развития информационного общества;
* готовность к самостоятельным поступкам и действиям, принятию ответственности за их результаты; готовность к осуществлению индивидуальной и коллективной информационной деятельности;
* способность к избирательному отношению к получаемой информации за счет умений ее анализа и критичного оценивания; ответственное отношение к информации с учетом правовых и этических аспектов ее распространения;
* развитие чувства личной ответственности за качество окружающей информационной среды;

**Индивидуализация обучения.** В учебном пособии имеются задания (всего их 63) разного уровня сложности. Это позволяет учителю построить для каждого учащегося индивидуальную образовательную траекторию.

**Методы оценивания уровня достижения учащегося:**

Обучение на высоком уровне трудности сопровождается соблюдением меры трудности, которая выражена в контроле качества усвоения. В систему проверки и контроля включены разнообразные способы контроля, но в любом случае система должна обладать развивающей по отношению к учащимся функцией. Для этого необходимо выполнение следующих условий:

* ни одно задание не должно быть оставлено без проверки и оценивания со стороны преподавателя;
* результаты проверки должны сообщаться незамедлительно;
* школьник должен максимально участвовать в процессе проверки выполненного им задания.

Главное в контроле - не оценка знаний и навыков посредством отметок, а дифференцированное и возможно более точное определение качества усвоения, его особенностей у разных учеников данного класса.

Практическая реализация принципа изучения в быстром темпе подразумевает постоянный контроль за знаниями и умениями учащихся, так как без убежденности в полном усвоении материала всеми учениками нет смысла двигаться вперед.

Текущий контроль уровня усвоения материала должен осуществляться, в основном, по результатам выполнения учащимися практических заданий на компьютере. Для грубой оценки можно воспользоваться результатами ответов на вопросы тестов, приведенных в конце каждой главы учебного пособия (ответить на эти же вопросы можно с помощью интерактивных веб-страниц, доступных на Microsoft-CD).

Итоговый контроль осуществляется по результатам защиты итоговых проектов. В начале курса каждому учащемуся нужно предложить в течение всего времени изучения курса разработать проект для решения некоторой задачи. В процессе защиты учащийся должен представить работающую компьютерную программу, которая решает поставленную перед ним задачу, и обосновать способ ее решения.

**Формы организации учебных занятий**

При изложении курса рекомендуется применять мини-лекции и компьютерный практикум. Учебный материал, в зависимости от специфики строящихся изображений, компонуется по блокам" каждый из которых охватывает одно из направлений компьютерной графики. Курс предваряет одночасовая лекция о компьютерной графике. В ходе этой лекции учитель повторяет с обучаемыми материал и через систему наводящих вопросов узнает о степени сформированности у них знаний по теме. Каждый блок начинается с мини-лекции: учитель выдает необходимый минимум теоретического материала (назначение и форматы операторов, особенности применяемых алгоритмов, способы построения изображений), снабженного демонстрационными примерами. Далее учащиеся получают задания (некоторые из которых, по возможности, должны были выполняться ими ранее с помощью пакетов компьютерной графики) и приступают к их выполнению на компьютерном практикуме. Подобный подход реализует один из важных методологических принципов — параллельное изложение со сравнением, что позволяет ученикам глубже постигать суть выполняемых операций. Занятия заканчиваются выдачей домашнего задания. Задания не должны быть очень сложными, и желательно, чтобы они носили творческий характер. Например, по рисунку и части программы, описывающей построение, дописать программу, или по исходной программе и части рисунка восстановить рисунок. Блок завершается выполнением творческого задания, а весь курс — проектной работой. При выполнении творческих заданий и проектной работы учащимся необходимо продемонстрировать сформированность знаний и умений по блокам курса.

**Учебно-тематическое планирование**

|  |  |  |
| --- | --- | --- |
| **Номер темы** | **Название темы** | **Кол-во часов** |
| **1** | Программы в повседневной жизни | 2 |
| **2** | Система программирования Visual Basic | 6 |
| **3** | Алгоритмы и программы | 6 |
| **4** | Формы и элементы управления | 6 |
| **5** | Свойства и методы | 6 |
| **6** | Присваивание и переменные | 6 |
| **7** | Операции | 6 |
| **8** | Ветвление: неполная форма | 6 |
| **9** | Ветвление: полная форма | 6 |
| **10** | Циклы со счетчиком | 6 |
| **11** | Циклы с условием | 6 |
| **12** | Подпрограммы и функции | 8 |
|  | **Всего** | **70** |

**Содержание элективного курса**

*Учебный модуль 1 «Программы в повседневной жизни»*

Программы в повседневной жизни. Чем занимаются программисты. Что такое программа. Возможности языков программирования. Синтаксис языков программирования

*Учебный модуль 2 «Система программирования Visual Basic»*

Visual Basic и IDE. Запуск и настройка Visual Studio.NET. Создание первого проекта. Конструирование графического интерфейса проекта. Создание программного кода проекта. Построение решения. Запуск проекта. Сохранение проекта. Вывод сообщений на форму

*Учебный модуль 3 «Алгоритмы и программы»*

Основные элементы кода. Алгоритм в форме псевдокода. Комментарии в коде

*Учебный модуль 4 «Формы и элементы управления»*

Форма — основа графического интерфейса. Свойства форм. Элементы управления и их свойства. Генерация событий

*Учебный модуль 5 «Свойства и методы»*

С чего начинается код. Чтение значений свойств в коде. Присваивание значений свойствам в коде. IntelliSense и точечная нотация. Методы

*Учебный модуль 6 «Присваивание и переменные»*

Присваивание. Переменные. Объявление переменных. Переменные в программах.

*Учебный модуль 7 «Операции»*

Арифметические операции. Строковые операции. Логические операции. Отладка кода

*Учебный модуль 8 «Ветвление: неполная форма»*

Булева логика. Операции сравнения. Оператор If...Then. Множественные условия. Булевы операции в коде

*Учебный модуль 9 «Ветвление: полная форма»*

Вложенные операторы If ...Then. Противоположные условия.Оператор If ...Then...Else. Пошаговое выполнение If. Операторы If в языках С# и J#. Булевы операции и операции сравнения в С# и J#

*Учебный модуль 10 «Циклы со счетчиком»*

Циклы For...Next. Пошаговое выполнение цикла

For...Next. Проекты с использованием For ...Next. Вложенные циклы. Выход из циклов. Циклы в С# и J#

*Учебный модуль 11 «Циклы с условием»*

Циклы Do While...Loop. Пошаговое выполнение цикла Do While...Loop. Циклы Do Until...Loop. Проекты с использованием Do...Loop. Циклы с постусловием. Циклы в С# и J#. Выход из циклов

*Учебный модуль 12 «Подпрограммы и функции»*

Что такое подпрограммы? Создание и вызов подпрограмм. Подпрограммы с аргументами. Создание и вызов собственных функций. Встроенные функции. Функции в J# и С#

**Методическая литература**

1. Абрамов С. А. Математические построения и программирование. М.: Наука, 1978.
2. Абрамов С. А., Гнездилова Г. Г., Капустина Е. К, Селюн М. И. Задачи по программированию. М.: Наука. Гл. ред. физ.-мат. лит., 1988.
3. Аляев Ю. А. Алгоритмизация и языки программирования Pascal,C++, Visual Basic: Справочное пособие. М.: Финансы и статистика, 2003.
4. Ананьева В. А. Геометрические построения // Информатика и образование. 2000.
5. Андрафанова Н. В. Построение графиков функций // Информатика и образование. 2000. № 6.
6. Бешенков С. А., Ракитина Е. А. Решение типовых задач по моделированию // Информатика в школе: Приложение к журналу «Информатика и образование». 2005. № 1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Дата проведения** | **№**  **п/п** | **Наименование темы, раздела**  Календарно-тематическое планирование | | **Примечание** |
| **Теория** | **Компьютерный практикум** |
|  |  | *Учебный модуль 1 «Программы в повседневной жизни»* | |  |
|  | 1.1 | Программы в повседневной жизни. |  |  |
|  | 1.2 | Чем занимаются программисты. |  |
|  | 1.3 | Что такое программа. |  |
|  | 1.4 | Возможности языков программирования. |  |
|  | 1.5 | Синтаксис языков программирования |  |
|  |  | **Контроль знаний и умений:** | Тест 1 |  |
|  |  | Учебное пособие: Глава 1. Программы в повседневной жизни.  Методическое пособие: 1. История развития языков программирования | Microsoft-CD: Глава 1. Программы вокруг нас |  |
|  |  | *Учебный модуль 2 «Система программирования Visual Basic»* | |  |
|  | 2.1 | Visual Basic .NET и IDE. | 1.Проект «Привет, Мир»  2.Проект «Мое имя» |  |
|  | 2.2 | Запуск и настройка Visual Studiо |  |
|  | 2.3 | Создание первого проекта. |  |
|  | 2.4 | Конструирование графического интерфейса проекта. |  |
|  | 2.5 | Создание программного кода проекта. |  |
|  | 2.6 | Построение решения. |  |
|  | 2.7 | Запуск проекта. |  |
|  | 2.8 | Сохранение проекта. |  |
|  | 2.9 | Вывод сообщений на форму |  |
|  |  | **Контроль знаний и умений:** | Тест 2 |  |
|  |  | Учебное пособие: Глава 2. Система программирования Visual Basic .NET.  Методическое пособие: 5. Подготовка учебного класса к занятиям | Microsoft-CD: Глава 2. Система программирования Visual Basic .NET |  |
|  |  | *Учебный модуль 3 «Алгоритмы и программы»* | |  |
|  | 3.1. | Основные элементы кода.  псевдокода | 1.Записать в форме псевдокода и построить блок-схему алгоритма включения компьютера, запуска операционной системы Windows и системы  программирования Visual Basic .NET.  2.Проект «Цена бензина».  3.Проект «Цвет рыбок» |  |
|  | 3.2. | Алгоритм в форме |  |
|  | 3.3. | Комментарии в коде |  |
|  |  | **Контроль знаний и умений:** | Тест 3 |  |
|  |  | Учебное пособие: Глава 3. Алгоритмы и программы.  Методическое пособие: Приложение 2, параграф 3. Основные типы алгоритмических структур и их кодирование на языке Visual Basic | Microsoft-CD: Глава 3. Алгоритмы и программы |  |
|  |  | *Учебный модуль 4 «Формы и элементы управления»* | |  |
|  | 4.1. | Форма — основа графического интерфейса. | 1.Проект «Замена шины»  2.Проект «Цена пиццы» |  |
|  | 4.2. | Свойства форм. |  |
|  | 4.3. | Элементы управления и их свойства. |  |
|  | 4.4. | Генерация событий |  |
|  |  | **Контроль знаний и умений:** | Тест 4 |  |
|  |  | Учебное пособие: Глава 4. Формы и элементы управления.  Методическое пособие: 2. Введение в объ  ектно-ориентированное визуальное программирование. 3. Краткий обзор .NET Framework и Visual Studio .NET. Приложение 1. Реализация объектно-ориентированного программирования в Visual Basic .NET | Microsoft-CD: Глава 4. Формы и элементы управления.  Таблица 1. Основы объектно-ориентированного программирования.  Таблица 2. Элементы управления, входящие в базовую поставку Visual Basic .NET: свойства, методы и события (начало).  Таблица 3. Элементы управления, входящие в базовую поставку Visual Basic .NET: свойства, методы и события (продолжение 1).  Таблица 4. Элементы управления, входящие в базовую поставку Visual Basic .NET: свойства, методы и события (продолжение 2).  Таблица 5. Элементы управления, входящие в базовую поставку Visual Basic .NET: свойства, методы и события (продолжение 3) |  |
|  |  | *Учебный модуль 5 «Свойства и методы»* | |  |
|  | 5.1. | С чего начинается код. | 1. Проект «Чтение свойств».  2. Проект «Установка свойств ».  3. Проект «Intellisence».  4. Проект «Методы» |  |
|  | 5.2. | Чтение значений свойств в коде. |  |
|  | 5.3. | Присваивание значений свойствам в коде. |  |
|  | 5.4. | IntelliSense и точечная нотация. |  |
|  | 5.5. | Методы |  |
|  |  | **Контроль знаний и умений:** | Тест 5 |  |
|  |  | Учебное пособие: Глава 5. Свойства и методы.  Методическое пособие: Приложение 2, параграф 5. Возможности работы с графикой в Visual Basic .NET 2003 и Visual Basic 2005 Express Edition | Microsoft-CD: Глава 5. Свойства и методы |  |
|  |  | *Учебный модуль 6 «Присваивание и переменные»* | |  |
|  | 6.1. | Присваивание. | 1. Проект «Прыжок лягушки».  2. Проект «Цвет». 3. Проект «Число Pi».  4. Проект «Количество щелчков».  5. Проект «Найди ошибки» |  |
|  | 6.2. | Переменные. |  |
|  | 6.3. | Объявление переменных. |  |
|  | 6.4. | Переменные в программах |  |
|  |  | **Контроль знаний и умений:** | Тест 6 |  |
|  |  | Учебное пособие: Глава 6. Присваивание и переменные. Методическое пособие: Приложение 2, параграф 1. Переменные в языке программирования Visual Basic .NET | Microsoft-CD: Глава 6. Присваивание и переменные. Таблица 6. Типы переменных |  |
|  |  | *Учебный модуль 7 «Операции»* | |  |
|  | 7.1. | Арифметические операции. | 1. Проект «Цена бензина-1».  2. Проект «Диаметр окружности».  3. Проект «Путь».  4. Проект «Цена бензина-2».  5. Проект «Имя, фамилия».  6. Проект «Логика».  7. Проект «Отладка».  8. Проект «Объем» |  |
|  | 7.2. | Строковые операции. |  |
|  | 7.3. | Логические операции. |  |
|  | 7.4. | Отладка кода |  |
|  |  | **Контроль знаний и умений:** | Тест 7 |  |
|  |  | Учебное пособие: Глава 7. Операции. Методическое пособие: Приложение 2, параграф 4. Функции в языке программиро  вания Visual Basic .NET | Microsoft-CD: Глава 7. Операции |  |
|  |  | *Учебный модуль 8 «Ветвление: неполная форма»* | |  |
|  | 8.1. | Булева логика. | 1. Проект «Логические операции».  2. Проект «If\_Then».  3. Проект «Пожарная тревога».  4. Проект «Выбор цвета».  5. Проект «Выбор цвета-2».  6. Проект «Выбор подарка».  7. Проект «Проверка» |  |
|  | 8.2. | Операции сравнения. |  |
|  | 8.3. | Оператор If...Then. |  |
|  | 8.4. | Множественные условия. |  |
|  | 8.5. | Булевы операции в коде |  |
|  |  | **Контроль знаний и умений:** | Тест 8 |  |
|  |  | Учебное пособие: Глава 8. Ветвление: не- полная форма.  Методическое пособие: Приложение 2, пункт 3.1. Алгоритмическая структура « ветвление » | Microsoft-CD: Глава 8. Ветвление: неполная форма.  Таблица 7. Алгоритмическая структура «ветвление» и ее кодирование на языке программирования |  |
|  |  | *Учебный модуль 9 «Ветвление: полная форма»* | |  |
|  | 9.1. | Вложенные операторы If ...Then. | 1. Проект «Флажки».  2. Проект «If Then Otherwise».  3. Проект «If-Then-Else».  4. Проект «If-Then-Else-2».  5. Проект «Гольф».  6. Проект «Step-In-If».  7. Проект «Магический квадрат».  8\* Проект «Бросок монеты» |  |
|  | 9.2. | Противоположные условия. |  |
|  | 9.3. | Оператор If ...Then...Else. |  |
|  | 9.4. | Пошаговое выполнение If. |  |
|  | 9.5. | Операторы If в языках С# и J#. |  |
|  | 9.6. | Булевы операции и операции сравнения в С# и J# |  |
|  |  | **Контроль знаний и умений:** | Тест 9 |  |
|  |  | Учебное пособие: Глава 9. Ветвление: полная форма.  Методическое пособие: Приложение 2, пункт 3.1. Алгоритмическая структура  «ветвление» | Microsoft-CD: Глава 9. Ветвление: полная форма.  Таблица 7. Алгоритмическая структура «ветвление» и ее кодирование на языке программирования |  |
|  |  | *Учебный модуль 10 «Циклы со счетчиком»* |  |  |
|  | 10.1. | Циклы For...Next. | 1.Проект «For-Next».  2.Проект «Сложение».  3.Проект «Сложение строк».  4.Проект «Цвет формы».  5.Проект «Подсчет кроликов»  6.Проект «Цикл в цикле».  7.Проект «Выход из циклов»  8.Проект «Простые числа»  9.Проект «Пенсия» |  |
|  | 10.2. | Пошаговое выполнение цикла  For...Next. |  |
|  | 10.3. | Проекты с использованием For ...Next |  |
|  | 10.4. | Вложенные циклы. |  |
|  | 10.5. | Выход из циклов. |  |
|  | 10.6. | Циклы в С# и J# |  |
|  |  | **Контроль знаний и умений:** | Тест 10 |  |
|  |  | Учебное пособие: Глава 10. Циклы со счетчиком.  Методическое пособие: Приложение 2, пункт 3.3. Алгоритмическая структура  « цикл » | Microsoft-CD: Глава 10. Циклы со счетчиком.  Таблица 9. Алгоритмическая структура «цикл со счетчиком» и ее кодирование на языке программирования |  |
|  |  | *Учебный модуль 11 «Циклы с условием»* | |  |
|  | 11.1. | Циклы Do While...Loop. | 1.Проект «Do-While-Loop».  2.Проект «Do-Until-Loop».  3.Проект «Do-While-Loop-2 ».  4.Проект «Do-Until-Loop-2».  5.Проект «Циклы с постусловием».  6.Проект «Exit-Do».  7.Проект «Мишень».  8.Проект «Монета».  9.Проект «Средняя оценка» |  |
|  | 11.2. | Пошаговое выполнение цикла Do  While...Loop. |  |
|  | 11.3. | Циклы Do Until...Loop. |  |
|  | 11.4. | Проекты с использованием Do...Loop |  |
|  | 11.5. | Циклы с постусловием. |  |
|  | 11.6. | Циклы в С# и J#. |  |
|  | 11.7. | Выход из циклов |  |
|  |  | **Контроль знаний и умений:** | Тест 11 |  |
|  |  | Учебное пособие: Глава 11. Циклы с условием.  Методическое пособие: Приложение 2, пункт 3.3. Алгоритмическая структура «цикл» | Microsoft-CD: Глава 11. Циклы с условием.  Таблица 10. Алгоритмическая структура «цикл с условием» и ее кодирование на языке программирования |  |
|  |  | *Учебный модуль 12 «Подпрограммы и функции»* | |  |
|  | 12.1. | Что такое подпрограммы? | 1.Проект «Подпрограмма».  2.Проект «Функция».  3.Проект «Единицы измерения».  4.Проект «Встроенные функции».  5.Проект «Предсказатель».  6.Проект «Кот и мышь» |  |
|  | 12.2. | Создание и вызов подпрограмм. |  |
|  | 12.3. | Подпрограммы с аргументами. |  |
|  | 12.4. | Создание и вызов собственных функций. |  |
|  | 12.5. | Встроенные функции. |  |
|  | 12.6. | Функции в J# и С# |  |
|  |  | **Контроль знаний и умений:** | Тест 12 |  |
|  |  | Учебное пособие: Глава 12. Подпрограммы и функции.  Методическое пособие: Приложение 1. Реализация объектно-ориентированного программирования в Visual Basic .NET | Microsoft-CD: Глава 12. Подпрограммы и функции |  |