

Учреждение образования  
«Белорусский государственный педагогический  
университет имени Максима Танка»

УТВЕРЖДАЮ

Проректор по учебной работе БГПУ

В.М.Зеленкевич

2019 г.

Регистрационный № УД-24-2-130-1<sup>2019</sup> уч.



## ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ И МЕТОДЫ АЛГОРИТМИЗАЦИИ

Учебная программа учреждения высшего образования  
по учебной дисциплине для специальности  
1-02 05 02 Физика и информатика

2019 г.

Учебная программа составлена на основе Образовательного стандарта высшего образования первая ступень специальность 1-02 05 02 Физика и информатика, утвержден и введен в действие постановлением Министерства образования РБ 30.08.2013 г. № 87 и учебного плана специальности 1-02 05 02 Физика и информатика.

#### **СОСТАВИТЕЛИ:**

С.В.Вабищевич, заведующий кафедрой информатики и методики преподавания информатики учреждения образования «Белорусский государственный педагогический университет имени Максима Танка», кандидат педагогических наук, доцент;

Г.А.Заборовский, доцент кафедры информатики и методики преподавания информатики учреждения образования «Белорусский государственный педагогический университет имени Максима Танка», кандидат физико-математических наук, доцент.

#### **РЕЦЕНЗЕНТЫ:**

В.М.Котов, заведующий кафедрой дискретной математики и алгоритмики учреждения образования «Белорусский государственный университет», доктор физико-математических наук, профессор;

В.В.Сидорик, доцент кафедры "Программное обеспечение информационных систем и технологий" Белорусского национального технического университета, кандидат физико-математических наук, доцент.

#### **РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:**


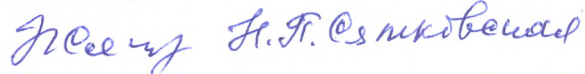
Кафедрой информатики и методики преподавания информатики (протокол № 11 от 23.05.2019);

Заведующий кафедрой  С.В.Вабищевич

Научно-методическим советом БГПУ (протокол № 6 от 18.06.2019).

Оформление учебной программы и сопровождающих ее материалов действующим требованиям Министерства образования Республики Беларусь соответствует

Методист учебно-методического отдела БГПУ

Директор библиотеки  С.А.Стародуб  
 Н.Ф.Шкубавич

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Учебная дисциплина «Технологии программирования и методы алгоритмизации» предназначена для студентов учреждений высшего образования, обучающихся по специальности 1–02 05 02 «Физика и информатика». Типовая учебная программа по учебной дисциплине «Технологии программирования и методы алгоритмизации» составлена в соответствии с требованиями образовательного стандарта Республики Беларусь и типового учебного плана этой специальности.

В современном информационном обществе значительно возрос спрос на специалистов в области информационных технологий, которые получают фундаментальные основы подготовки в этой области при изучении школьного курса информатики. Учебная дисциплина «Технологии программирования и методы алгоритмизации» играет ключевую роль в системе профессиональной подготовки преподавателей информатики, являясь основой и неотъемлемым компонентом их профессиональной деятельности. Будущему преподавателю информатики необходимо знать основные парадигмы и технологии программирования, что подразумевает овладение несколькими языками программирования. Это позволяет быть конкурентоспособным и мобильным как в системе образования, так и в профессиональной сфере в целом. Учебная дисциплина базируется на усвоенном ранее студентами учебном материале учебной дисциплины «Компьютерная графика и мультимедиа».

*Целью преподавания учебной дисциплины* является формирование профессиональных компетенций преподавателя физики и информатики в области технологий программирования и методов алгоритмизации.

*Задачи изучения дисциплины:*

- формирование знаний о современных парадигмах, системах, языках и технологиях программирования;
- обучение использованию различных алгоритмических методов при решении задач из разных предметных областей;
- формирование практических навыков разработки программных продуктов в различных современных средах программирования.

### Требования к освоению учебной дисциплины

В результате изучения дисциплины обучаемый должен:

**знать:**

- основные технологии, языки и системы программирования;
- структуры данных, базовые структуры алгоритмов, методы анализа алгоритмов;
- методы использования типовых алгоритмов в решении практических задач;
- принципы и технологии структурного и объектно-ориентированного программирования;
- методы и этапы разработки и отладки программ;

**уметь:**

- работать в современных системах программирования;
- реализовывать в компьютерной среде базовые алгоритмы;
- осуществлять сравнительный анализ языков программирования;
- разрабатывать объектные модели в различных предметных областях;
- создавать приложения на основе объектных моделей;

**владеть:**

- способами проектирования алгоритмов и их реализацией с помощью современных средств программирования;
- навыками разработки, отладки и тестирования программ, написанных на объектно-ориентированном языке высокого уровня;
- приемами использования современных технологий программирования в профессиональной деятельности учителя физики и информатики.

**Требования к освоению учебной дисциплины в соответствии с образовательным стандартом**

Согласно образовательному стандарту высшего образования ОСВО 1-02 05 02 - 2013 специальности 1-02 05 02 Физика и информатика изучение учебной дисциплины «Информационные технологии в образовании» должно обеспечить формирование у студентов академических, социально-личностных и профессиональных компетенций.

***Требования к академическим компетенциям***

Специалист должен:

- АК-1. Уметь применять базовые научно-теоретические знания для решения теоретических и практических задач.
- АК-2. Владеть методами научно-педагогического исследования.
- АК-4. Уметь работать самостоятельно.
- АК-5. Быть способным порождать новые идеи (обладать креативностью).
- АК-7. Иметь навыки, связанные с использованием технических устройств, управлением информацией и работой с компьютером.
- АК-9. Уметь учиться, повышать свою квалификацию в течение всей жизни.

***Требования к социально-личностным компетенциям***

Специалист должен:

- СЛК-4. Владеть навыками здоровьесбережения.
- СЛК-6. Уметь работать в команде.
- СЛК-7. Быть способным к осуществлению самообразования и самосовершенствования профессиональной деятельности.

***Требования к профессиональным компетенциям***

Специалист должен быть способен:

Обучающая деятельность

- ПК-1-О. Эффективно реализовывать обучающую деятельность.
- ПК-6-О. Организовывать самостоятельную работу обучающихся.

Развивающая деятельность

– ПК-2-Р. Развивать навыки самостоятельной работы обучающихся с учебной, справочной, научной литературой и др. источниками информации.

Ценностно-ориентационная деятельность

– ПК-4-Ц. Осуществлять самообразование и самосовершенствование профессиональной деятельности.

### **Распределение общего количества часов по семестрам.**

Трудоёмкость учебной дисциплины составляет 9 зачётных единиц.

Дневная форма получения образования.

Всего на учебную дисциплину «Технологии программирования и методы алгоритмизации» учебным планом предусмотрено 338 часов, из них 150 часов аудиторных занятий (58 лекционных, 92 лабораторных).

Текущая аттестация проводится в соответствии с учебным планом специальности в форме зачёта (3 семестр), экзаменов (4 и 5 семестр).

Семестр	Всего	Количество аудиторных часов			Количество часов самостоятельной работы	Форма контроля	Зачётных единиц
		Лекции	Лабораторные занятия	Всего			
3	64	10	24	34	30	зачёт	2
4	132	26	32	58	38	экзамен(36)	3
5	142	22	36	58	48	экзамен(36)	4
Всего	338	58	92	150	116	72	9

### **Структура содержания учебной дисциплины.**

Дисциплина изучается на протяжении трех семестров и содержит три раздела. В первом разделе изучаются базовые алгоритмы и методы структурно-модульного программирования; во втором – технологии объектно-ориентированного программирования, в третьем рассматривается развитие языков и технологий программирования. Особое внимание уделяется применению различных технологий программирования при решении практических задач в контексте специализации и будущей педагогической деятельности студентов.

Изучение структурно-модульного программирования предлагается реализовать на языке Pascal (Pascal ABC). Для обучения технологиям объектно-ориентированного программирования выбран язык C#, который реализуется на платформе .Net в среде Microsoft Visual Studio (C# Develop). В завершение рассматривается язык программирования Python. Такой выбор обусловлен направленностью на школьную информатику и соответствует принципу преемственности при изучении технологий программирования и методов алгоритмизации, а также способствует развитию общей культуры программирования.

### **Методы и формы обучения.**

Обучение учебной дисциплине проходит в рамках организации лекционных и лабораторных занятий. В лекционном курсе рассматриваются новейшие концепции и подходы к программированию, обсуждаются возможности и особенности языка и выбранного программного обеспечения. Обращается внимание на методологию постановки и решения задач, выбор и анализ алгоритмов, эффективность программ. При чтении лекций особое внимание следует уделять демонстрации реальных программных продуктов и мультимедийным презентациям, которые должны служить для будущих учителей образцом объяснения материала. Электронные учебно-методические комплексы обеспечивают студентов необходимыми теоретическими и справочными материалами.

Лабораторные занятия направлены на формирование навыков практического использования полученных знаний при выполнении конкретных заданий. Методика их проведения должна содействовать развитию индивидуально-творческих способностей каждого студента и приобретению навыков самостоятельной работы. Акцент сделан на практической работе с системами программирования. С целью подготовки будущего учителя к решению задач информатизации сферы образования рекомендуется предусмотреть задания по разработке компьютерных программ образовательного назначения.

Текущий контроль осуществляется при выполнении и сдаче лабораторных работ. Наряду с традиционными методами и формами обучения целесообразно использовать приемы проблемного и частично-поискового методов обучения, новые формы: веб-занятие, коллективная работа в сети др. Важную роль играет исследовательская работа студентов. Эта работа должна органично включаться в учебный процесс со всеми видами занятий.

Освоение системы знаний и умений по учебной дисциплине «Технологии программирования и методы алгоритмизации» позволит студентам продуктивно осваивать учебный материал по таким учебным дисциплинам, как «Информационные системы и сети», «Методика преподавания информатики», «Информационные технологии в образовании».

## СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

### РАЗДЕЛ 1. СТРУКТУРНО-МОДУЛЬНОЕ ПРОГРАММИРОВАНИЕ

#### **Тема 1.1. Основы технологии программирования.**

Понятие технологии программирования. Методология решения задач с помощью компьютера. Модульный принцип в программировании. Отладка и тестирование программ. Ручная отладка, вывод промежуточных результатов. Тестирование на контрольных примерах. Документирование программ. Оценки качества программ.

#### **Тема 1.2. Языки и системы программирования.**

Роль и характеристики языков программирования. История развития языков программирования. Классификации языков программирования. Проблема универсального языка программирования и универсальной вычислительной машины. Основные понятия языков программирования. Компилируемые и интерпретируемые языки. Стандартизация графического представления алгоритмов. Состав и назначение систем программирования. Компиляторы и интерпретаторы. Языки школьной информатики.

#### **Тема 1.3. Язык программирования Pascal. Базовые алгоритмические конструкции.**

Концепция структурного программирования. Отражение идей структурного программирования в языке Pascal. Состав языка. Основные понятия. Переменные, операции, выражения. Процедуры и функции. Структура программы. Типы данных. Ввод/вывод данных.

Базовые алгоритмические конструкции структурного программирования: следование, ветвление, цикл. Алгоритмы целочисленной арифметики. Нахождение наибольшего общего делителя, наименьшего общего кратного. Поиск чисел с заданными свойствами (простых, палиндромов, и др...) Разложение чисел на простые множители. Перевод чисел из одной системы счисления в другую. Делимость чисел. Действия с многозначными числами. Задачи комбинаторики. Работа с вещественными числами. Анализ алгоритмов.

#### **Тема 1.4. Графические возможности языка Pascal.**

Графические примитивы. Процедуры и функции для работы с графикой. Действия с пером и кистью. Работа с цветом. Действия со шрифтом Действия с графическим окном. Действия с рисунками. Использование растровых изображений. Анимация.

#### **Тема 1.5. Составные типы данных.**

Массивы: описание, способы ввода и вывода элементов массива. Одномерные и многомерные массивы: Типовые задачи обработки массивов. Методы работы с элементами массивов. Алгоритмы сортировки. Символы. Процедуры и функции работы с символами. Строки. Процедуры и функции работы со строковыми данными. Типовые задачи обработки строк. Множества. Описание. Операции над множествами. Примеры работы с числовыми и символьными множе-

ствами. Записи. Описание. Примеры работы с записями. Организация обработки массивов записей.

### **Тема 1.6. Механизмы структурирования программ.**

Процедуры и функции как средство структурирования программ. Создание процедур и функций пользователя: правила записи и вызова. Локальные и глобальные переменные. Параметры процедур и функций, обмен данными с основной программой. Рекурсия. Создание и использование модулей. Библиотеки пользователя.

### **Тема 1.7. Алгоритмы работы с внешней и внутренней памятью компьютера.**

Файловые типы. Обмен данными с диском и внешними устройствами. Типизированные, текстовые, нетипизированные файлы. Указатели и динамические структуры данных: списки, стеки, очереди.

### **Тема 1.8. Событийно-ориентированное программирование.**

Парадигмы программирования. Событийно-ориентированное программирование в среде Pascal. События клавиатуры и мыши. Виртуальные коды клавиш. Обработка событий. Объектно-ориентированное программирование в среде Pascal. Классы и объекты. Описание класса. Состав класса. Процедуры и функции как методы класса. Расширение графических возможностей языка Pascal ABC. Проектирование графического интерфейса. Новые возможности программирования в среде Pascal ABC.Net.

## **РАЗДЕЛ 2. ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ**

### **Тема 2.1. Платформа .Net Framework. Система MS Visual Studio .Net.**

Архитектура платформы .Net Framework. Виртуальная машина. Общезыковая исполнительная среда CLR платформы .Net Framework. Трансляция программного кода в .Net. Библиотеки классов. Пространства имен. Сборка, решение, проект. Системы и технологии быстрой разработки программ (RAD). Среды визуального проектирования программного обеспечения (IDE). Система Microsoft Visual Studio .Net. Работа в среде MS Visual Studio (C# Develop). Интерфейс. Окна. Основные инструменты. Разработка консольных приложений.

### **Тема 2.2. Язык программирования C#.**

Основные понятия языка C#. Переменные. Операторы. Выражения. Синтаксис. Ввод и вывод данных. Типы данных. Типы - как классы .Net Framework. Преобразования типов. Базовые алгоритмические конструкции: ветвление if, выбор switch, циклы while, for, foreach. Работа с целыми и вещественными числами. Классы Math, Random.

### **Тема 2.3. Ссылочные типы данных.**

Массивы. Класс Array. Одномерные, многомерные, прямоугольные, ступенчатые массивы. Формирование массивов. Работа с массивами. Сортировка и поиск в массивах. Операции с векторами и матрицами.

Исключительные ситуации. Обработка исключений.



Символы и строки. Класс Char. Работа с символами. Классы String и StringBuilder. Работа со строками. Регулярные выражения. Особенности использования регулярных выражений при обработке текстов.

Потоки ввода-вывода. Работа с файлами. Запись и чтение. Классы StreamWriter и StreamReader.

#### **Тема 2.4. Основы объектно-ориентированного программирования.**

Понятие класса и объекта. Состав класса. Модификаторы элементов класса. Поля. Методы. Статические классы. Статические поля и методы. Параметры методов. Вызов методов и способы передачи параметров. Перегрузка методов. Создание и инициализация объекта. Конструкторы. Организация доступа к полям класса. Свойства. Инкапсуляция. Скрытие полей, создание свойств.

Принципы ООП. Реализация ООП в среде MS Visual Studio. Этапы разработки программ. Рефакторинг. Визуальное представление классов. Основы UML. Конструирование классов с помощью визуального дизайнера.

#### **Тема 2.5. Наследование. Полиморфизм.**

Разновидности наследования. Классическое наследование. Модель включения – делегирования. Композиция и агрегация. Стратегии наследования. Виртуальные методы. Реализации полиморфизма. Переопределение методов. Использование виртуальных методов. Раннее и позднее связывание. Параметризованные типы. Бесплодные классы. Абстрактные классы. Иерархия классов. Класс object. Интерфейсы. Реализации интерфейсов. Стандартные интерфейсы .Net. Делегаты. События. Достоинства и недостатки ООП.

### **РАЗДЕЛ 3. РАЗВИТИЕ СИСТЕМ, ЯЗЫКОВ И ТЕХНОЛОГИЙ ПРОГРАММИРОВАНИЯ**

#### **Тема 3.1. Проектирование приложений с графическим интерфейсом в среде MS Visual Studio.**

Объектно-событийная модель работы Windows-приложения. Проектирование Windows-приложений. Графический интерфейс. Элементы управления Windows.Forms. Класс Control. Свойства, методы, события. Класс Form. Свойства, методы, события. События мыши и клавиатуры. Этапы создания Windows-приложения. Структура проекта. Разделение визуального представления и функциональности. Использование элементов TextBox, Button, Label, CheckBox, RadioButton, PictureBox, TrackBar, NumericUpDown, ListBox, ComboBox. Основные свойства и методы. Обработка событий.

#### **Тема 3.2. Создание приложений Windows.Forms.**

Разработка простого текстового редактора. Разработка графического интерфейса. Организация меню. Диалоговые окна. Работа с папками и файлами. Работа с датой и временем. Класс Timer. Использование таймера. Анимация. Работа с базами данных.

Работа с графикой. GDI. Классы и методы рисования. Класс Graphic. Классы Pen, Brush. Рисование фигур. Разработка простого графического редактора. Обработка событий мыши и клавиатуры.

### **Тема 3.3. Создание приложений на основе Windows Presentation Foundation.**

Основы XAML. Унификация интерфейсов. Элементы управления. Графические возможности WPF. Работа со шрифтами. Работа с растровой графикой. Анимация. Звук и видео.

Использование возможностей среды MS Visual Studio C# для разработки образовательных средств. Разработка простых компьютерных моделей и демонстраций. Системы тестирования.

### **Тема 3.4. Перспективные языки и технологии программирования.**

Тенденции развития языков и систем программирования. Эволюция парадигм и технологий программирования.

Использование визуальных языков и систем для обучения программированию. Проблемно-ориентированные языки. Языки и технологии web-программирования. Языки и технологии инженерии знаний. Языки машинного обучения и искусственного интеллекта.

### **Тема 3.5. Основы языка программирования Python.**

Возможности и особенности языка программирования Python. Основные понятия. Синтаксис. Переменные. Операторы. Выражения. Типы данных. Интерактивный режим работы.

Структурно-модульное программирование на языке Python. Динамическая типизация. Работа со списками и строками. Кортежи. Словари. Базовые алгоритмические конструкции. Использование библиотек Python.

Разработка приложений с графическим интерфейсом. Графические библиотеки. Функциональное программирование на языке Python. Объектно-ориентированное программирование на языке Python. Возможности использования языка Python в науке и образовании.

## УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Номер раздела, темы, занятия	Название раздела, темы, занятия; перечень изучаемых вопросов	Количество аудиторных часов		Самостоятельная работа студента	Материальное обеспечение занятия (наглядные, методические пособия и др)	Литература	Форма контроля знаний
		Лекции	Лабораторные занятия				
1	2	3	4	5	6	7	8
<b>3 семестр</b>							
<b>1.</b>	<b>1. Структурно-модульное программирование (48 ч.)</b>	<b>18</b>	<b>30</b>				
<b>1.1.</b>	<b>Основы технологии программирования (2 ч.)</b>	<b>2</b>		<b>2</b>			
1.1.1.	Понятие технологии программирования. Методология решения задач с помощью компьютера. Модульный принцип в программировании. Отладка и тестирование программ. Тестирование на контрольных примерах. Документирование программ. Оценки качества программ.	2		2	Компьютерная презентация	1,3	Входной тест
<b>1.2.</b>	<b>Языки и системы программирования (2 ч.)</b>	<b>2</b>		<b>2</b>			
1.2.1.	Роль и характеристики языков программирования. История развития языков программирования. Классификации языков программирования. Основные понятия языков программирования. Компилируемые и интерпретируемые языки. Стандартизация графического представления алгоритмов. Состав и назначение систем программирования. Компиляторы и интерпретаторы. Языки школьной информатики.	2		2	Компьютерная презентация	1,3,4	Тест
<b>1.3.</b>	<b>Язык программирования Pascal. Базовые алгоритмические конструкции (12 ч.)</b>	<b>2</b>	<b>10</b>	<b>10</b>			
1.3.1.	Концепция структурного программирования. Отражение идей структурного программирования в языке Pascal. Состав языка. Основные понятия. Переменные, операции, выражения. Процедуры и функции.	2	2	2	Электрон. сопровождение лекций (ЭСЛк)	1,2,3	Тест
1.3.2.	Структура программы. Типы данных. Ввод/вывод данных. Базовые конструкции структурного программирования: следование, ветвление,		2 2	2			

1.3.3	цикл. Алгоритмы целочисленной арифметики. Нахождение наибольшего общего делителя, наименьшего общего кратного. Поиск чисел с заданными свойствами (простых, палиндромов, и др...)		2	2	Инструкции к лаб. работам		Тест Контрольные задания, тесты Рейтинговая работа
1.3.4	Разложение чисел на простые множители. Перевод чисел из одной системы счисления в другую. Делимость чисел. Действия с многозначными числами. Задачи комбинаторики.		2	2			
1.3.5	Работа с вещественными числами. Анализ алгоритмов.			2			
<b>1.4.</b>	<b>Графические возможности языка Pascal (6 ч.)</b>	<b>2</b>	<b>4</b>	<b>6</b>			
1.4.1.	Графические примитивы. Процедуры и функции для работы с графикой.	2		2	ЭСЛк Инструкции к лаб. работам	1,4,6	Контрольные задания, тесты
1.4.2.	Действия с пером и кистью. Работа с цветом. Действия со шрифтом		2	2			
1.4.3	Действия с графическим окном. Действия с рисунками. Использование растровых изображений. Анимация.		2	2			
<b>1.5.</b>	<b>Составные типы данных (12 ч.)</b>	<b>2</b>	<b>10</b>	<b>10</b>			
1.5.1.	Массивы: описание, способы ввода и вывода элементов массива. Одномерные и многомерные массивы: Символы. Строки.	2	2	2	Презентация ЭСЛк	1,4,6	Контрольные задания, тесты Рейтинговая работа
1.5.2.	Типовые задачи обработки массивов. Методы работы с элементами массивов. Алгоритмы сортировки.		2	2			
1.5.3.	Процедуры и функции работы с символами. Процедуры и функции работы со строковыми данными. Типовые задачи обработки строк.		2	2	Инструкции к лаб. работам		
1.5.4.	Множества. Операции над множествами. Записи.		2	2			
1.5.5	Примеры работы с числовыми и символьными множествами. Примеры работы с записями. Организация обработки массивов записей.		2	2			
<b>Всего за 3 семестр</b>		<b>10</b>	<b>24</b>	<b>30</b>			<b>Зачёт</b>
<b>4 семестр</b>							
<b>1.6.</b>	<b>Механизмы структурирования программ (4ч.)</b>	<b>2</b>	<b>2</b>	<b>6</b>			
1.6.1.	Процедуры и функции как средство структурирования программ.	2	2	2	Презентация Инструкции к лаб. работам	1,4,5	Контрольные задания, тесты
1.6.2.	Создание процедур и функций пользователя: правила записи и вызова.			2			
1.6.3.	Локальные и глобальные переменные. Параметры процедур и функций, обмен данными с основной программой. Рекурсия.			2			
1.6.4.	Создание и использование модулей. Библиотеки пользователя.						
<b>1.7.</b>	<b>Алгоритмы работы с внешней и внутренней памятью компьютера</b>	<b>2</b>	<b>2</b>	<b>6</b>			

	<b>(4 ч.)</b>						
1.7.1.	Файловые типы. Типизированные, текстовые, нетипизированные файлы.	2		2	Презентация	1,3,6	Контроль-ные задания Рейтинговая работа
1.7.2.	Обмен данными с диском и внешними устройствами.		2	2	Инструкции к лаб. работам		
1.7.3.	Указатели и динамические структуры данных: списки, стеки, очереди.			2			
<b>1.8.</b>	<b>Событийно-ориентированное программирование (6 ч.)</b>	<b>4</b>	<b>2</b>	<b>6</b>			
1.8.1.	Парадигмы программирования. Событийно-ориентированное программирование.	2	2	2	Презентация	5,6	Контроль-ные задания Коллоквиум
1.8.2.	Объектно-ориентированное программирование в среде Pascal. Классы и объекты. Состав класса. Процедуры и функции как методы класса. Обработка событий. События клавиатуры и мыши. Виртуальные коды клавиш.			2	Инструкции к лаб. работам		
1.8.3.	Расширение графических возможностей языка Pascal ABC. Проектирование графического интерфейса. Новые возможности программирования в среде Pascal ABC.Net.	2		2			
<b>2.</b>	<b>Объектно-ориентированное программирование (50 ч.)</b>	<b>20</b>	<b>32</b>				
<b>2.1.</b>	<b>Платформа .Net Framework. Система MS Visual Studio.Net (8ч.)</b>	<b>4</b>	<b>4</b>	<b>6</b>			
2.1.1.	Архитектура платформы .Net Framework. Виртуальная машина. Общезыкоуая исполнительная среда CLR. Трансляция программного кода в .Net. Библиотеки классов. Пространства имен.	2		2	Электрон. сопровождение лекций (ЭСЛк) Инструкция к лаб. работе	9, 10	Контроль-ные задания Рейтинговая работа
	Системы и технологии быстрой разработки программ (RAD). Среды визуального проектирования программного обеспечения (IDE). Система Microsoft Visual Studio .Net.	2	2	2			
2.1.2.	Работа в среде MS Visual Studio (C# Develop). Интерфейс. Окна. Основные инструменты. Разработка консольных приложений.		2	2			
<b>2.2.</b>	<b>Язык программирования C# (10 ч.)</b>	<b>4</b>	<b>6</b>	<b>4</b>			
2.2.1.	Основные понятия языка C#. Переменные. Операторы. Выражения. Синтаксис. Ввод и вывод данных.	2		2	Презентация.	8, 9	Контроль-ные задания Рейтинговая работа
2.2.2.	Типы данных. Типы - как классы .Net Framework. Преобразования типов. Базовые алгоритмические конструкции.	2		2	ЭСЛк.		
2.2.3.	Реализация алгоритмов ветвления и выбора: if, switch.		2		Инструкции к лаб. работам		
2.2.4.	Реализация циклических алгоритмов: while, for, foreach.		2				
2.2.5.	Работа с целыми и вещественными числами. Классы Math, Random.		2				
2.2.6.			2				
<b>2.3.</b>	<b>Ссылочные типы данных (14 ч.)</b>	<b>4</b>	<b>10</b>	<b>4</b>			

2.3.1.	Массивы. Класс Array. Одномерные, многомерные, прямоугольные, ступенчатые массивы. Сортировка и поиск в массивах.	2		2	Презентации ЭСЛк.	8, 9	Контроль- ные задания
2.3.2.	Формирование массивов. Работа с одномерными массивами. Работа с прямоугольными и ступенчатыми массивами.		2				
2.3.3.	Исключительные ситуации. Обработка исключений.		2		Инструк- ции к лаб. работам		Рейтинговая работа
2.3.4.	Символы и строки. Класс Char. Классы String и StringBuilder. Регуляр- ные выражения.	2		2			
2.3.5.	Работа с символами.			2			
2.3.6.	Работа со строками. Использование регулярных выражений. Потоки ввода-вывода. Классы StreamWriter и StreamReader.		2	2			
2.3.7.	Работа с файлами. Запись и чтение.		2				
<b>2.4.</b>	<b>Основы объектно-ориентированного программирования (12 ч.)</b>	<b>6</b>	<b>6</b>	<b>6</b>			
2.4.1.	Понятие класса и объекта. Состав класса. Модификаторы элементов класса. Поля. Методы. Статические классы, поля и методы Параметры методов. Вызов методов и способы передачи параметров.	2		2	Презента- ции ЭСЛк. Инструк- ции к лаб. работам	8, 9, 12	Контроль- ные задания Коллоквиум
2.4.2.	Создание классов. Создание и инициализация объектов.		2				
2.4.3.	Создание конструкторов. Перегрузка методов.	2	2	2			
2.4.4.	Организация доступа к полям класса. Свойства. Инкапсуляция. Соккрытие полей, создание свойств.			2			
2.4.5.	Принципы ООП. Реализация ООП в среде MS Visual Studio. Этапы разработки программ. Рефакторинг. Визуальное представление клас- сов. Основы UML.	2		2			
2.4.6.	Конструирование классов с помощью визуального дизайнера.		2				
<b>Всего за 4 семестр</b>		<b>26</b>	<b>32</b>	<b>38</b>			<b>Экзамен</b>
<b>5 семестр</b>							
<b>2.5.</b>	<b>Наследование. Полиморфизм (8 ч.)</b>	<b>2</b>	<b>6</b>	<b>4</b>			
2.5.1	Разновидности наследования. Классическое наследование. Модель включения – делегирования. Композиция и агрегация. Стратегии наследования. Виртуальные методы. Раннее и позднее связывание. Параметризованные типы. Бесплодные классы.	2	2	2	Презента- ции. ЭСЛк.	8, 9, 12	Контроль- ные задания Рейтинговая работа
2.5.2	Наследование. Реализация полиморфизма. Переопределение методов. Использование виртуальных методов.		2				
2.5.3	Абстрактные классы. Иерархия классов. Класс object. Интерфейсы.		2				

	Реализации интерфейсов. Стандартные интерфейсы .Net. Делегаты. События. Достоинства и недостатки ООП.			2			
--	---	--	--	---	--	--	--

РЕПОЗИТОРИЙ БГПУ

<b>3.</b>	<b>Развитие систем, языков и технологий программирования (50 ч.)</b>	<b>20</b>	<b>30</b>				
<b>3.1.</b>	<b>Проектирование приложений с графическим интерфейсом в среде MS Visual Studio (8 ч.).</b>	<b>4</b>	<b>4</b>	<b>10</b>			
3.1.1.	Объектно-событийная модель работы Windows-приложения. Проектирование Windows-приложений. Графический интерфейс. Элементы управления Windows.Forms. Класс Control. Свойства, методы, события. Класс Form. Свойства, методы, события.	2		2	Презентация. ЭСЛк.	10, 12	
3.1.2.	Этапы создания Windows-приложения. Структура проекта. Разделение визуального представления и функциональности. События мыши и клавиатуры.	2		2	Демонстрации		
3.1.3.	Использование элементов TextBox, Button, Label, CheckBox, RadioButton, PictureBox. Основные свойства и методы. Обработка событий.		2	2	Инструкция к лаб. работе		Контрольные задания, тест
3.1.4.	Разработка интерактивных приложений. Использование элементов TrackBar, NumerickUpDown, ListBox, ComboBox.		2	2			
<b>3.2.</b>	<b>Создание приложений Windows.Forms (12 ч.)</b>	<b>4</b>	<b>8</b>	<b>12</b>			
3.2.1.	Разработка графического интерфейса. Организация меню. Диалоговые окна. Работа с датой и временем.	2		2	Презентация. ЭСЛк.	10, 12	
3.2.2.	Разработка простого текстового редактора. Использование диалоговых окон. Работа с папками и файлами.		2	2	Демонстрации		Контрольные задания, тест
3.2.3.	Класс Timer. Использование таймера. Анимация.		2	2			Рейтинговая работа
3.2.4.	Работа с базами данных.		2	2	Инструкция к лаб. работам		
3.2.5.	Работа с графикой. GDI. Классы и методы рисования. Класс Graphic.	2					
3.2.6.	Классы Pen, Brush. Обработка событий мыши и клавиатуры. Рисование фигур. Разработка простого графического редактора.		2	2			
<b>3.3.</b>	<b>Создание приложений на основе Windows Presentation Foundation. (8 ч.)</b>	<b>4</b>	<b>4</b>	<b>10</b>			
3.3.1.	Графические возможности WPF. Основы XAML. Унификация интерфейсов. Элементы управления. Звук и видео. Анимация.	2		2	ЭСЛк. Демонстрации	9, 12	Экспресс-контроль
3.3.2.	Работа со шрифтами. Работа с растровой графикой.		2	2			
3.3.3.	Использование возможностей среды MS Visual Studio для разработки образовательных средств.	2		2			
3.3.4.	Разработка простых компьютерных моделей и демонстраций. Системы		2	2	Инструкция к лаб.		Контроль-



3.3.5.	тестирования.				работам		ные задания
<b>3.4.</b>	<b>Перспективные языки и технологии программирования (6 ч.).</b>	<b>2</b>	<b>4</b>	<b>4</b>			
3.4.1.	Тенденции развития языков и систем программирования. Эволюция парадигм и технологий программирования. Языки и технологии инженерии знаний. Языки машинного обучения и искусственного интеллекта. Проблемно-ориентированные языки.	2		2	ЭСЛк. Демонстрации	3, 11	Контрольные задания Рейтинговая работа
3.4.2.	Использование визуальных языков и систем для обучения программированию.		2		Инструкции к лаб. работам		
3.4.3.	Языки и технологии web-программирования.		2	2			
<b>3.5.</b>	<b>Основы языка программирования Python (16 ч.).</b>	<b>6</b>	<b>10</b>	<b>8</b>			
3.5.1.	Возможности и особенности языка программирования Python. Основные понятия. Синтаксис. Переменные. Операторы. Выражения. Типы данных. Интерактивный режим работы.	2		2	Презентации. ЭСЛк.	11	Контрольные задания Рейтинговая работа Коллоквиум
3.5.2.	Структурно-модульное программирование на языке Python. Динамическая типизация.		2	2	Инструкции к лаб. работам		
3.5.3.	Работа со списками и строками. Кортежи. Словари.		2				
3.5.4.	Базовые алгоритмические конструкции.		2				
3.5.5.	Использование библиотек Python.	2		2			
3.5.6.	Разработка приложений с графическим интерфейсом. Графические библиотеки		2				
3.5.7.	Функциональное программирование на языке Python. Объектно-ориентированное программирование на языке Python.		2		Демонстрации		
3.5.8.	Возможности использования языка Python в науке и образовании.	2		2			
	<b>Всего за 5 семестр</b>	<b>22</b>	<b>36</b>	<b>48</b>			<b>Экзамен</b>
	<b>Всего по учебной дисциплине</b>	<b>58</b>	<b>92</b>	<b>116</b>			

## ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

### ЛИТЕРАТУРА

#### Основная:

1. Вабищевич, С. В. Рабочая тетрадь по программированию на языке Паскаль / С. В. Вабищевич; М-во образования Респ. Беларусь, Белорус. гос. пед. ун-т. – Минск : БГПУ, 2018. – 42 с.
2. Вабищевич, С. В. УМК по учебной дисциплине «Технологии программирования и методы алгоритмизации» [Электронный ресурс]. – Режим доступа: <http://elib.bspu.by/handle/doc/6915>. – Дата доступа: 14.04.2019.
3. Городко, С. И. Современные технологии программирования : учеб.-метод. пособие для студентов / С. И. Городко, С. В. Снисаренко ; М-во образования Респ. Беларусь, Белорус. гос. ун-т информатики и радиоэлектроники, Фак. компьютер. проектирования. – Минск : БГУИР, 2017. – 67 с.
4. Кутыш, А. З. Взаимосвязное обучение технологиям программирования : практикум для студентов учреждений высш. образования, обучающихся по специальности 1-02 05 01 Математика и информатика / А. З. Кутыш ; М-во образования Респ. Беларусь, Белорус. гос. пед. ун-т. – Минск : БГПУ, 2017. – 200 с.

#### Дополнительная:

5. Алгоритмы: построение и анализ / Т. Х. Кормен, [и др.]. – М. : Вильямс, 2013. – 1328 с.
6. Окулов, С. М. Программирование в алгоритмах / С. М. Окулов. – М.: БИНОМ, 2013. – 384 с.
7. Медведик, В. Практика программирования на Паскаль. Задачи и решения / В. Медведик. – М. : ДМК-Пресс, 2015. – 590 с.
8. Павловская, Т. А. С#. Программирование на языке высокого уровня / Т. А. Павловская. – СПб. : Питер, 2014. – 432 с.
9. Подбельский, В. В. Язык С#. Базовый курс / В. В. Подбельский. – М. : Финансы и статистика, 2015. – 408 с.
10. Культин, Н. Б. Основы программирования в Microsoft Visual C# 2010. – СПб. : БХВ-Петербург, 2014. – 368 с.
11. Фёдоров, Д. Ю. Основы программирования на примере языка Python / Д. Ю. Фёдоров. – СПб. : БХВ-Петербург, 2018. – 164 с.
12. Флёнов, М. Библия С# / М. Флёнов. – СПб. : БХВ-Петербург, 2016. – 544 с.

## ТЕМАТИЧЕСКИЙ ПЛАН

№	Наименование раздела, темы	Всего	Лек- ции	Лабора- торные занятия
<b>1</b>	<b>Структурно-модульное программирование</b>	<b>48</b>	<b>18</b>	<b>30</b>
1.1	Основы технологии программирования	2	2	
1.2	Языки и системы программирования	2	2	
1.3	Язык программирования Pascal. Базовые алгоритмические конструкции	12	2	10
1.4	Графические возможности языка Pascal	6	2	4
1.5	Составные типы данных	12	2	10
1.6	Механизмы структурирования программ	4	2	2
1.7	Алгоритмы работы с внешней и внутренней памятью компьютера	4	2	2
1.8	Событийно-ориентированное программирование	6	4	2
<b>2</b>	<b>Объектно-ориентированное программирование</b>	<b>52</b>	<b>20</b>	<b>32</b>
2.1	Платформа .NET Framework. Система MS Visual Studio .Net	8	4	4
2.2	Язык программирования С#	10	4	6
2.3	Ссылочные типы данных	14	4	10
2.4	Основы объектно-ориентированного программирования.	12	6	6
2.5	Наследование. Полиморфизм	8	2	6
<b>3</b>	<b>Развитие систем, языков и технологий программирования</b>	<b>50</b>	<b>20</b>	<b>30</b>
3.1	Проектирование приложений с графическим интерфейсом в среде MS Visual Studio.	8	4	4
3.2	Создание приложений Windows.Forms.	12	4	8
3.3	Создание приложений на основе Windows Presentation Foundation.	8	4	4
3.4	Перспективные языки и технологии программирования.	6	2	4
3.5	Основы языка программирования Python.	16	6	10
<b>Всего:</b>		<b>150</b>	<b>58</b>	<b>92</b>

## **МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ И ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ**

Содержание и формы самостоятельной работы студентов разрабатываются в соответствии с целями и задачами подготовки специалиста. Для управления самостоятельной работой рекомендуется использовать:

- среды программирования (Pascal ABC, Microsoft Visual Studio, C# Develop, IDLE Python и др...);
- электронные средства обучения (презентации, демонстрации готовых примеров компьютерных программ и моделей);
- работу с электронным ресурсным центром;
- тестирующие программы.

Текущий контроль осуществляется в ходе выполнения и защиты лабораторных работ. Самостоятельная работа студента методически организуется путем выполнения заданий по материалу каждой лабораторной работы.

Особое внимание необходимо обращать на организацию индивидуальной работы студента под руководством преподавателя. Эта работа должна проводиться с учётом индивидуальных особенностей каждого студента с помощью системы индивидуальных заданий, которые студент может выполнять на основе образцов.

## ТРЕБОВАНИЯ К ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТА

№ п/п	Название раздела, темы,	К-во часов СРС	Задание	Форма выполнения
<b>1.</b>	<b>Структурно-модульное программирование</b>	<b>48</b>		
1.1.	Основы технологии программирования	2	Отладка и тестирование программ. Тестирование на контрольных примерах.	Пример документированной программы
1.2.	Языки и системы программирования	2	История развития языков программирования. Классификации языков программирования.	Обзор Отчёт о самостоятельной работе
1.3.	Язык программирования Pascal. Базовые алгоритмические конструкции	10	Использование языка Pascal в школе. Решение задач на реализацию базовых конструкций структурного программирования. Решение задач целочисленной арифметики.	Примеры алгоритмов.  Коды программ
1.4.	Графические возможности языка Pascal	6	Решение типовых задач программирования графики	Коды программ
1.5.	Составные типы данных	10	Решение задач с использованием массивов. Решение задач с использованием символов и строк. Решение задач с использованием записей и множеств.	Примеры алгоритмов. Коды программ
1.6.	Механизмы структурирования программ	6	Создание процедур и функций. Решение задач с использованием собственных процедур и функций	Примеры алгоритмов. Коды программ
1.7.	Алгоритмы работы с внешней и внутренней памятью компьютера	6	Работа с файлами. Работа с динамическими структурами	Примеры алгоритмов. Коды программ
1.8.	Событийно-ориентированное программирование	6	Проектирование графического интерфейса. Создание программы учебного назначения с использованием обработки событий.	Проект интерфейса и код программы учебного назначения
<b>2.</b>	<b>Объектно-ориентированное программирование</b>	<b>24</b>		
2.1.	Платформа .Net Framework. Система MS Visual Studio.Net	6	Сравнение систем разработки программного обеспечения.	Обзор

2.2.	Язык программирования C#	4	Преобразования типов в C#. Решение задач на реализацию базовых алгоритмических конструкций.	Таблица преобразований. Коды программ. Примеры алгоритмов.
2.3.	Ссылочные типы данных	4	Решение задач с использованием массивов. Решение задач с использованием символов и строк. Решение задач с регулярными выражениями	Примеры алгоритмов. Коды программ. Шаблоны выражений.
2.4.	Основы объектно-ориентированного программирования.	6	Создание классов и объектов. Решение задач с перегрузкой методов. Инкапсуляция. Защита данных. Конструирование классов с помощью визуального дизайнера.	Коды программ. Алгоритмы защиты. Схемы классов.
2.5.	Наследование. Полиморфизм	4	Решение задач с использованием наследования. Использование виртуальных методов. Решение задач с классами сложной структуры	Схемы классов. Коды программ
<b>3.</b>	<b>Развитие систем, языков и технологий программирования</b>	<b>44</b>		
3.1.	Проектирование приложений с графическим интерфейсом в среде MS Visual Studio	10	Проектирование интерфейсов Windows-приложений. Использование элементов управления.	Проект интерфейса. Обзор компонентов.
3.2.	Создание приложений Windows.Forms.	12	Разработка приложений учебного назначения (текстовый или графический редактор, система тестирования, компьютерная модель физического явления)	Описание разработки. Рабочая версия приложения.
3.3.	Создание приложений на основе Windows Presentation Foundation.	10	WPF-приложение с элементами мультимедиа. Разработка образовательных средств с использованием WPF.	Описание разработки. Рабочая версия приложения.
3.4.	Перспективные языки и технологии программирования.	4	Эволюция парадигм программирования. Эволюция языков и технологий программирования	Обзор. Обзор.
3.5.	Основы языка программирования Python.	8	Использования языка Python в науке и образовании. Решение задач на реализацию базовых алгоритмов. Разработка программ учебного назначения	Обзор. Примеры алгоритмов. Коды программ
	<b>Всего</b>	<b>116</b>		

## **ПЕРЕЧЕНЬ ИСПОЛЬЗУЕМЫХ СРЕДСТВ ДИАГНОСТИКИ РЕЗУЛЬТАТОВ УЧЕБНОЙ ДЕЯТЕЛЬНОСТИ**

Для оценки достижений и уровня знаний студента при изучении дисциплины целесообразно применить комплексный инструментарий, который включает:

- входной контроль перед выполнением лабораторной работы;
- контроль выполнения заданий лабораторной работы;
- контроль выполнения заданий для самостоятельной работы по каждой лабораторной работе;
- контроль выполнения самостоятельной работы по темам;
- отчеты о самостоятельной работе;
- экспресс контроль по темам;
- компьютерное тестирование по разделам и темам;
- коллоквиумы по разделам и темам;
- рейтинговые оценки прохождения дисциплины в семестре.

**Протокол согласования рабочей программы  
с другими дисциплинами специальности**

Название учебной дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола)
Компьютерная графика и мультимедиа	Кафедра математики и методики преподавания математики	Предусмотреть изучение основ математической логики	Протокол № 11 от 23.05.2019

РЕПОЗИТОРИЙ БГПУ