

1. Цели изучения учебной дисциплины.

ЦЕЛЬ - познакомить студентов с содержанием и основными задачами современных технологий разработки программного обеспечения (ПО), а также с возможностями их применения, встречающихся в современной науке и технике.

2. Место учебной дисциплины в структуре образовательной программы.

Учебная дисциплина входит в базовую часть блока 1, изучается в 1 и 2 семестрах.

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОП.

В рамках изучения дисциплины обеспечивается формирование следующих компетенций:

- владением широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий (ОПК-1);
- способностью выбирать и оценивать способ реализации информационных систем и устройств (программно, аппаратно или программно-аппаратно) для решения поставленной задачи (ОПК-6);
- способностью разрабатывать средства реализации информационных технологий (методические, информационные, математические, алгоритмические, технические и программные) (ПК-12);
 - способностью адаптировать приложения к изменяющимся условиям функционирования (ПК-32).

Студент, освоивший программу, должен:

1. *Обладать профессиональными компетенциями*, включающими в себя способность:
 - использовать технологии программирования в профессиональной деятельности;
 - решать задачи воспитания и духовно нравственного развития личности;
 - применять современные методики и технологии для обеспечения качества учебно-воспитательного процесса.
2. *Знать и уметь* использовать в своей профессиональной деятельности:
 - Основные этапы производства программного продукта, принципы их построения, приемы работы с инструментальными средствами, поддерживающими разработку программного обеспечения.
 - Теоретические основы построения современных языков программирования, инструментальных и технологических средств разработки ПО.
 - Реализовать выбор технологии и инструментальных средств, для разработки, составления, отладки, тестирования и документирования ПО на языках высокого уровня для задач обработки числовой и символьной информации.
 - Способы записи алгоритмов и конструирования программ с использованием различных алгоритмических языков.

3. Владеть навыками процедурного и объектно-ориентированного программирования

4. Общая трудоемкость дисциплины 12 зачетных единиц и виды учебной работы.

Вид учебной работы	Трудоемкость (в соответствии с учебным планом) (час)	Распределение по семестрам (в соответствии с учебным планом) (час)			
	432 (в том числе в интер. – 40)	1 (в том числе в интер. – 20)	2 (в том числе в интер. – 20)		
Аудиторные занятия	202	76	126		
Лекции	80	38	42		
Практические занятия					
Семинары					
Лабораторные работы	122	38	84		
Другие виды аудиторных работ					
Другие виды работ					
Самостоятельная работа	176	80	96		
Курсовой проект (работа)					
Реферат					
Расчётно-графические работы					
Формы текущего контроля					
Формы промежуточной аттестации в соответствии с учебным планом	54	экзамен 27	экзамен 27		

5. Содержание учебной дисциплины.

5.1. Разделы (темы) учебной дисциплины

№п/п	Наименование раздела дисциплины (темы)	Аудиторные часы					Самостоятельная работа (час)
		ВСЕГО	Лекции	Практические (семинары)	Лабораторные работы	В т.ч. интерактивные формы обучения (не менее 10%)	
1	Понятие о программировании. Парадигмы программирования.	4	4				2
2	Введение в программирование на языке Паскаль.	40	20		20	2	32
3	Работа с внешней памятью в языке Паскаль.	12	6		6	2	10
4	Работа с динамической памятью.	12	6		6	4	10
5	Использование структур данных.	12	6		6	4	12
6	Основы ООП.	12	12				10

№п/п	Наименование раздела дисциплины (темы)	Аудиторные часы					Самостоятельная работа (час)
		ВСЕГО	Лекции	Практические (семинары)	Лабораторные работы	В т.ч. интерактивные формы обучения (не менее 10%)	
7	Object Pascal, как язык объектно-ориентированного программирования. Понятие о визуальном проектировании.	4	4				8
8	Библиотека визуальных компонентов.	18	4		14	4	16
9	Создание приложений на языке Object Pascal.	32	8		24	4	20
10	Ресурсы, буфер обмена, печать, таймеры.	6			6	4	10
11	Библиотеки динамической компоновки	6			6	4	10
12	Логическое и функциональное программирование	14	4		10	4	12
13	Web-программирование	18	2		16	4	12
14	CASE-технологии	12	4		8	4	12
	Итого:	202/5,6 зач.ед	80	-	122	40/19,8%	176

5.2. Содержание разделов дисциплины.

1. Понятие о программировании. Парадигмы программирования.

Алгоритмизация и программирования. Программа для ЭВМ, как средство реализации алгоритма. Высокоуровневое и низкоуровневое программирование. Парадигмы высокоуровневого программирования: процедурная, объектно-ориентированная, функциональная, рекурсивно-логическая. Понятие о структурном программировании.

2. Введение в программирование на языке Паскаль.

Структура программы. Линейные программы. Алфавит языка. Простые и составные типы данных. Операторы ветвления и циклов. Процедуры и функции.

Использование массивов. Одномерные и многомерные массивы. Сортировка массивов. Работа с векторами и матрицами.

Множества. Записи. Оператор присоединения. Записи с вариантами.

Понятие рекурсии. Рекурсивные алгоритмы.

Описания типов. Пользовательские типы. Интервальный и перечислимый типы.

3. Работа с внешней памятью в языке Паскаль.

Файлы. Типизированные, нетипизированные и текстовые файлы. Работа с содержимым файла. Работа с файлами на уровне операционной системы.

4. Работа с динамической памятью.

Статическое и динамическое выделение памяти. Указатели. Процедуры `getmem` и `freemem`, `new` и `dispose`. Динамическое выделение памяти под массив. Понятие о динамических структурах данных. Организация односвязных и двусвязных линейных списков, деревьев. Тип `Pointer`.

5. Использование структур данных.

Понятие о стеках и очередях, область их применения. Моделирование стеков и очередей на массивах и списках.

Использование бинарных деревьев для эффективного поиска информации. Принцип сбалансированности. AVL-деревья.

6. Основы ООП.

Языки программирования и системы программирования. Понятие парадигмы. Процедурно-ориентированная парадигма, ее особенности, процедурные языки, принцип разделения данных и процедур, Паскаль – как язык структурного программирования. Понятие объекта, как структуры содержащей данные и процедуры – 2 подхода. Манифест ОО-систем. Свойства и методы объектов, классы и типы, свойства и методы класса и экземпляра, конструкторы и деструкторы. Инкапсуляция. Сообщения и события. Наследование, понятие абстрактных классов, иерархия или сеть классов. Обработка событий. Понятие визуального проектирования. Визуальное проектирование и ООП. Обзор основных объектно-ориентированных языков программирования: SmallTalk, C++, Java, Object Pascal.

7. Object Pascal, как язык объектно-ориентированного программирования. Понятие о визуальном проектировании.

Object Pascal, как результат эволюционных изменений языка Pascal. Типы данных языка Pascal, специфичные для Windows. Объекты и классы, ссылочная модель объекта, объявление класса, методы Create и Free – как конструктор и деструктор. Поддержка инкапсуляции – ключевые слова private, public и protected. Классы и модули. Ключевое слово Self. Методы и данные класса. Указатели на методы, как развитие идеи процедурного типа. Наследование, наследование и совместимость с типов. Виртуальные и динамические методы. Информация о типе на этапе выполнения. Отладка программы. Пошаговое выполнение, точки останова, просмотр значений переменных на этапе выполнения. Создание консольных приложений.

Проблема быстрой разработки интерфейса пользователя и попытки ее решения с помощью объектно-ориентированной идеологии (Turbo Vision, Windows GUI). Обзор современных средств визуального.

8. Библиотека визуальных компонентов.

Иерархия классов, компоненты и объекты, их использование. Визуальные и не визуальные компоненты. Стандартные свойства компонентов. События, как свойства.

9. Создание приложений на языке Object Pascal.

Меню: главное меню, всплывающие меню. Кнопки, радиокнопки, чекбоксы, использование панелей. Окна ввода: компоненты Tedit и TspinEdit, списки, компонент TcomboBox. Компоненты TMemo и TRichEdit. Создание простейшего текстового редактора. Формы и окна, стили форм, автоматическое и ручное создание форм, размещение и размеры форм, формы при различных разрешениях экрана.

Графические компоненты в Lazarus, Timage и TPaintBox. Использование объекта TCanvas. Создание простейшего графического редактора. Создание панели инструментов (ToolBar) и строки состояния (StatusBar). Многодокументный интерфейс, создание MDI-приложений. Использование TOLEContainer. Стандартные диалоговые окна в Lazarus. Работа с внешними файлами в Lazarus.

10. Ресурсы, буфер обмена, печать, таймеры.

Использование ресурсов: курсоры, пиктограммы. Объект TPrinter. Буфер обмена, работа с буфером обмена. Объект Ttimer.

11. Библиотеки динамической компоновки.

Статическая и динамическая компоновка, библиотеки динамической компоновки, использование dll в приложениях Lazarus. Создание библиотеки dll. Отладка Dll.

12. Логическое и функциональное программирование.

Теоретические основы логического и функционального программирования. Элементы исчисления предикатов. Понятие о функциональном программировании.

13. Web-программирование.

Язык HTML как средство создания информационных ресурсов Интернет. Использование скриптов на стороне сервера (PHP, ASP, Perl) и на стороне клиента (JavaScript).

14. CASE-технологии

CASE-технология разработки ПО. Основные компоненты CASE-систем. Классификация CASE-средств. Понятие структурного системного анализа, как основного средства CASE-технологии. Диаграммы потоков данных.

5.3. Лабораторный практикум

№ п/п	№ раздела дисциплины	Наименование лабораторных работ
1.	2	Итерационные вычисления. Сумма элементов одномерного массива. Поиск максимального элемента. Сортировка массива простыми методами. Использование двумерных массивов. Использование множеств. Использование записей. Процедуры и функции. Рекурсии.
2.	3	Работа с файлами.
3.	4	Указатели. Динамическое выделение памяти. Динамические массивы. Линейные списки.
4.	5	Деревья. Организация стека и очереди на массивах. Организация стека и очереди на списках.
5.	8	Проектирование интерфейса пользователя
6.	9	Создание новых классов
7.	10	Использование кнопок. Использование компонента TListBox. Использование компонента TComboBox. Использование компонента TEdit. Использование компонента TMemo.
8.	11	Разработка приложений
9.	12	Решение логических задач
10.	13	Создание простейших Web-приложений
11.	14	Документирование ПО с помощью пакета CASE

6. Учебно-методическое обеспечение самостоятельной работы по дисциплине.

6.1. Основная литература по дисциплине:

1. Иванова Г. С. Технология программирования. - М.:КНОРУС, 2011. – 333 с.

6.2. Дополнительная литература:

1. Архангельский А.Я. Delphi 2006. – М.: Бинوم, 2006.
2. Архангельский А.Я. Язык Pascal и основы программирования в Delphi. – М.: Бинوم, 2004.

3. Братко И. Программирование на языке ПРОЛОГ для искусственного интеллекта. М: Мир, 1990. – 560 с.
4. Вирт Н. Системное программирование. Введение. – М: Мир, 1977.
5. Дантеманн Дж, Мишел Дж, Тейлор Д. Программирование в среде Delphi. – Киев: DiaSoft Ltd – 1995.
6. Дж. Донован. Системное программирование. – М: Мир, 1975. – 200 с.
7. Калянов Г.И. CASE. Структурный системный анализ (автоматизация и применение). – М: Лори, 1996.
8. Канту М. Delphi 2 для Windows 95/NT. Полный курс. Тома 1, 2. Москва. Внешторгиздат. 1996 г.
9. Климов Ю. С., Касаткин А. И., Мороз С. М. Программирование в среде Turbo Pascal 6.0. – Минск: Высшая школа. – 1992.
10. Коварцев А.Н. Автоматизация разработки и тестирования программных средств. Самар. гос. аэрокосм. ун-т. Самара, 1999. – 150 с.
11. Культин Н.Б. Программирование в Turbo Pascal 7.0 и Delphi. – СПб.: BHV – Санкт-Петербург, 1998. – 240 с.
12. Липаев В.В. Отладка сложных программ. Методы, средства, технология. - М.: Энергоатомиздат, 1993.
13. Липаев В.В. Системное проектирование сложных программных средств для информационных систем. - М.: СИНТЕГ, 1999. - 224 с.
14. Рубенкинг Н. Программирование в Delphi для "чайников". - К.: "Диалектика", 1996. – 304 с.
15. Турский В. Методология программирования. – М: Мир. 1981.

6.3. Перечень ресурсов информационно-телекоммуникационной сети Интернет (далее- сеть Интернет), необходимых для освоения дисциплины.

Учебно-методические пособия и разработки по технологии программирования и программированию на сайтах

1. <http://intuit.ru>
2. <http://linux.armd.ru/ru/documentation/metod/> - методические пособия по ПСПО

6.4. Рекомендации по использованию информационных технологий, включая перечень программного обеспечения и информационных справочных систем.

№п/п	Наименование раздела (темы) учебной дисциплины (модуля)	Наименование материалов обучения, пакетов программного обеспечения	Наименование технических и аудиовизуальных средств, используемых с целью демонстрации материалов
1.	1-5	FreePascal	Проектор
2.	8-14	Lazarus, Object Pascal	Проектор

7. Методические рекомендации по организации изучения дисциплины.

7.1. Методические рекомендации для студентов:

Самостоятельная работа студента предполагает различные формы индивидуальной учебной деятельности: конспектирование научной литературы, сбор и анализ практического материала в СМИ, проектирование, выполнение тематических и творческих заданий. Выбор форм и видов самостоятельной работы определяется индивидуально-личностным подходом к обучению совместно преподавателем и студентом.

8. Формы оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Вопросы и задания для самостоятельной работы, в том числе групповой самостоятельной работы обучающихся;

Решение задач в соответствии с лабораторным практикумом.

Перечень вопросов для промежуточной аттестации

Экзамен, 1 семестр:

1. Алгоритмы и программы. Языки программирования, их классификация (парадигмы).
2. Элементарные алгоритмические структуры.
3. Идентификаторы. Константы. Типы данных.
4. Стандартные типы данных. Перечислимый и интервальный типы данных.
5. Выражения и операции.
6. Структура программы на языке Паскаль.
7. Безусловный переход. Ветвления и выбор.
8. Операторы организации циклов и управления циклом.
9. Одномерные и многомерные массивы.
10. Записи. Доступ к полю. Оператор присоединения.
11. Множества и их внутреннее устройство.
12. Строки. Обработка строк.
13. Файлы. Типы файлов. Основные процедуры и функции работы с файлами.
14. Особенности работы с текстовыми файлами.
15. Особенности работы с типизированными и нетипизированными файлами. Последовательный и прямой доступ к содержимому файла.
16. Подпрограммы. Процедуры и функции. Их описание.
17. Описание параметров подпрограмм. Внутренний механизм организации вызовов.
18. Рекурсия и ее внутренне устройство.

Экзамен, 2 семестр:

1. Библиотечные модули. Их структура. Модули CRT и Graph.
2. Статическое и динамическое распределение памяти. Указатели.
3. Динамическое выделение и освобождение памяти. Динамическое выделение памяти под массив.
4. Линейные списки: односвязные и двусвязные.
5. Накопители данных: стеки и очереди. Их моделирование на стеках и массивах.
6. Деревья и их использование. AVL-деревья.
7. Основные принципы ООП: инкапсуляции, наследование и полиморфизм.
8. Реализация некоторых идей ООП в Turbo Pascal (Free Pascal).
9. Основные понятия ООП (свойства, методы, классы, наследование).
10. Понятие о визуальном проектировании.
11. Понятие проекта, его состав. Файлы, входящие в проект.
12. Простые типы данных в языке Object Pascal.
13. Составные типы данных в языке Object Pascal.
14. Продвинутое типы данных в языке Object Pascal (variant, указатели, процедурные типы).
15. Выражения и операции в языке Object Pascal.
16. Операторы в языке Object Pascal (:=, goto, простой и составной операторы, вызов процедуры).
17. Операторы ветвления и циклов в языке Object Pascal.

18. Процедуры и функции в языке Object Pascal. Локальные и глобальные переменные. Рекурсия.
19. Особенности ООП на языке Object Pascal. Инкапсуляция, наследование, полиморфизм.
20. Поля, свойства и методы объектов в языке Object Pascal.
21. Функция SendMessage.
22. Динамическая информация о типе, ее использование.
23. Иерархия классов,
24. Timer – генератор событий.
25. Особенности модальных форм. Управление диалоговыми формами в Object Pascal.
26. Процедуры и функции Object Pascal, реализующие простейшие диалоги.
27. Динамическая настройка меню средствами Object Pascal.
28. Объекты TApplication и TScreen в Object Pascal.
29. Технология OLE
30. Библиотеки динамической компоновки DLL, их использование.
31. Теоретические основы логического программирования.
32. Понятие о функциональном программировании.
33. Основные базовые принципы объектно-ориентированного программирования.
34. Использование скриптов на стороне сервера (PHP, ASP, Perl) и на стороне клиента (JavaScript)
35. Основные компоненты CASE-систем.
36. Классификация CASE-средств.
37. Понятие структурного системного анализа, как основного средства CASE-технологии.
38. Диаграммы потоков данных.

Рабочая программа учебной дисциплины составлена в соответствии с учебным планом, федеральным государственным образовательным стандартом высшего образования по направлению подготовки **09.03.02 – информационные системы и технологии.**

Рабочая программа учебной дисциплины составлена:

д.ф.-м.н., профессор кафедры информатики Горчаков Л.В. Горчаков

Рабочая программа учебной дисциплины утверждена на заседании кафедры информатики

протокол № 1 от « 31 » августа 2015 г.

Зав. кафедрой информатики Стась А.Н. Стась

Рабочая программа учебной дисциплины одобрена методической комиссией физико-математического факультета

протокол № 1 от « 31 » августа 2015 г.

Председатель методической комиссии Скрипко З.А. Скрипко