


МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Томский государственный педагогический университет»
(ТГПУ)

УТВЕРЖДАЮ
Декан физико-математического факультета

Е.Г. Пьяных, к.п.н., доцент

«26» мая 2016 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ (МОДУЛЯ)
ПРОГРАММИРОВАНИЕ

Направление подготовки: *44.03.05 Педагогическое образование (с двумя профилями подготовки)*

Направленности (профили): *Математика и Информатика*

Форма обучения: *очная*

1. Место учебной дисциплины (модуля) в структуре образовательной программы

Учебная дисциплина относится к вариативной части блока 1 (обязательные дисциплины).

Дисциплины, предшествующие изучению данной дисциплины: «Программное обеспечение ЭВМ» (часть 1).

Изучение данной дисциплины необходимо для успешного освоения следующих дисциплин: «Теоретические основы прикладной математики и информатики», «Архитектура компьютера», «Практикум по решению задач на ЭВМ».

2. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОП

Дисциплина обеспечивает формирование следующей компетенции:

✓ готовность реализовывать образовательные программы по учебным предметам в соответствии с требованиями образовательных стандартов (ПК-1).

Обучающийся, освоивший программу, должен:

1. *Знать и уметь*, а также 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.
- Способы записи алгоритмов и конструирования программ с использованием различных алгоритмических языков.

2. *Владеть* технологиями процедурного программирования.

3. Содержание учебной дисциплины (модуля)

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 невидимые 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.

4. Трудоемкость дисциплины (модуля) по видам учебных занятий, самостоятельной работы обучающихся и формам контроля

4.1. Очная форма обучения

Объем в зачетных единицах: 10.

4.1.1. Виды учебных занятий, самостоятельная работа обучающихся, формы контроля (в академических часах)

Вид учебной работы	Всего часов	Распределение по семестрам (в академических часах)	
		2	3
Лекции	56	38	18
Лабораторные работы	96	58	38
Практические занятия (семинары)			
Самостоятельная работа	154	93	61
Курсовая работа			
Другие виды занятий			
Формы текущего контроля		тест	тест
Формы промежуточной аттестации	54	27 (экзамен)	27 (экзамен)
Итого часов	360	216	144

4.1.2. Содержание учебной дисциплины (модуля), структурированное по темам (разделам)

№п/п	Наименование темы (раздела) дисциплины	Всего часов	Аудиторные занятия (в часах)			Самостоятельная работа (в часах)
			Лекции	Практические занятия (семинары)	Лабораторные работы	
1	Понятие о программировании. Парадигмы программирования.	23	6			17
2	Введение в программирование на языке Паскаль.	39	6		16	17
3	Работа с внешней памятью в языке Паскаль.	39	6		16	17
4	Работа с динамической памятью.	39	6		16	17
5	Использование структур данных.	39	6		16	17
6	Основы ООП.	23	6			17
7	Object Pascal, как язык объектно-ориентированного программирования. Понятие о визуальном проектировании.	23	6			17
8	Библиотека визуальных компонентов.	39	6		16	17
9	Создание приложений на языке Object Pascal.	42	8		16	18
	Итого:	306	56		96	154

4.1.3. Лабораторный практикум

№ п/п	Наименование темы (раздела)	Наименование лабораторных работ
-------	-----------------------------	---------------------------------

	ДИСЦИПЛИНЫ	
1.	Введение в программирование на языке Паскаль.	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.	Работа с внешней памятью в языке Паскаль.	Работа с файлами.
3.	Работа с динамической памятью.	Указатели. Динамическое выделение памяти. Динамические массивы. Линейные списки.
4.	Использование структур данных.	Деревья. Организация стека и очереди на массивах. Организация стека и очереди на списках.
5.	Библиотека визуальных компонентов.	Проектирование интерфейса пользователя.
6.	Создание приложений на языке Object Pascal.	Разработка приложений.

5. Перечень учебно-методического обеспечения по дисциплине (модулю)

5.1. Основная учебная литература

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

5.2. Дополнительная литература:

1. Архангельский А.Я. Delphi 2006. – М.: Бином, 2006.
2. Архангельский А.Я. Язык Pascal и основы программирования в Delphi. – М.: Бином, 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. Донован Дж. 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. – М: 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. . – Самара: Самар. гос. аэрокосм. ун-т, 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.

5.3. Перечень ресурсов информационно-коммуникационной сети Интернет, необходимых для освоения дисциплины (модуля)

Учебно-методические пособия и разработки по технологии программирования и программированию:

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

5.4. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)

№п/п	Номера разделов (тем) 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.	6-9	Lazarus, Object Pascal	Проектор

6. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Для проведения занятий необходимы: лекционная аудитория, кабинет программирования и информационных систем.

7. Методические указания для обучающихся по освоению дисциплины (модуля)

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 и пр. Выбор форм и видов самостоятельной работы определяется индивидуально-личностным подходом к обучению совместно преподавателем и студентом.

8. Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине (модулю)

Представлен в виде отдельного документа (приложение к рабочей программе учебной дисциплины (модуля)).

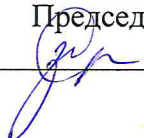
Рабочая программа учебной дисциплины (модуля) составлена в соответствии с учебным планом, федеральным государственным образовательным стандартом высшего образования по направлению подготовки 44.03.05 Педагогическое образование (с двумя профилями подготовки).

Рабочая программа учебной дисциплины (модуля) составлена
А.Н. Стаем, к.т.н., зав. кафедрой информатики

Рабочая программа учебной дисциплины (модуля) утверждена на заседании кафедры информатики
Протокол №10 от «26» мая 2016 года

Зав. кафедрой информатики _____  А.Н. Стась, к.т.н.

Рабочая программа учебной дисциплины (модуля) одобрена методической комиссией физико-математического факультета
Протокол № 9 от « 26 » мая 2016 года

Председатель учебно-методической комиссии физико-математического факультета
 З.А. Скрипко, д.п.н, профессор