

МИНИСТЕРСТВО ОБРАЗОВАНИЯ АЗЕРБАЙДЖАНСКОЙ РЕСПУБЛИКИ
Дербентский филиал Общества с ограниченной ответственностью
«Азербайджанский Государственный Экономический Университет»

Утверждаю
Ректор, профессор
_____ Мурадов А.Д.
« _____ » _____ 2017 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
ОП.06 Основы алгоритмизации и программирования

Специальность
09.02.04 Информационные системы (по отраслям)

Квалификация
техник по информационным системам

Программа подготовки
базовая

Форма обучения
очная

Рецензент: Мехтиев М.А. – доцент филиала

Рабочая программа предназначена для преподавания обще-профессиональной дисциплины профессионального цикла студентам очной формы обучения по специальности 09.02.04 Информационные системы (по отраслям).

Рабочая программа составлена с учетом Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.04 Информационные системы (по отраслям) Утвержденного Приказом Министерства образования и науки Российской Федерации от 14.05.2014 № 525

Составитель _____ Вурдиханов В.Р. – доцент филиала

Содержание

1. Цель и задачи освоения дисциплины	4
2. Место дисциплины в структуре ППСЗ	4
3. Требования к результатам освоения содержания дисциплины	4
4. Структура и содержание дисциплины	6
4.1. Объем учебной дисциплины и виды учебной работы	6
4.2. Тематический план учебной дисциплины	6
4.3. Содержание разделов (тем) дисциплины	7
4.4. Лабораторные работы (занятия)	9
4.5. Самостоятельное изучение тем (вопросов) дисциплины	9
5. Образовательные технологии	10
6. Оценочные средства для текущего контроля успеваемости и промежуточной аттестации (образцы)	10
7. Учебно-методическое обеспечение дисциплины	18
8. Материально-техническое обеспечение дисциплины	18

1. Цели и задачи освоения дисциплины

Курс «Основы алгоритмизации и программирования» является профессиональным, формирующим математический и информационный уровень знаний для освоения других профессиональных дисциплин.

Цели: изучение методов программирования для овладения знаниями в области технологии программирования, подготовка к использованию как языков программирования, так и методов программирования; получения основ и практических умений и навыков выработки и внедрения эффективных решений в области программирования.

Задачи: обзор и исследование основных понятий алгоритмов программирования на основе структурного, модульного и объектно-ориентированного и:

- знакомство с методами структурного и объектно-ориентированного программирования как наиболее распространенными и эффективными методами разработки программных продуктов следования;
- знакомство с основными структурами данных и типовыми методами обработки этих структур.

2. Место дисциплины в структуре ППСЗ

Предшествующие курсы, на которых непосредственно базируется дисциплина «Основы алгоритмизации и программирования» являются:

- «Устройство и функционирование информационной системы»;
- «Операционные системы и среды».

Вместе с тем курс «Основы алгоритмизации и программирования» является общепрофессиональным, устанавливающим базовый уровень знаний для освоения других общепрофессиональных и специальных дисциплин таких, как Основы проектирования баз данных, ПМ.01 Эксплуатация и модификация информационных систем, МДК.01.02 Методы и средства проектирования информационных систем.

3. Требования к результатам освоения содержания дисциплины

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС СПО и ППСЗ по данной специальности:

а) общих (ОК):

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.

ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.

ОК 5. Использовать информационно-коммуникационные технологии в профессиональной деятельности.

ОК 6. Работать в коллективе и команде, эффективно общаться с коллегами, руководством, потребителями.

ОК 7. Брать на себя ответственность за работу членов команды (подчиненных), результат выполнения заданий.

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

б) профессиональных (ПК):

ПК 1.2. Взаимодействовать со специалистами смежного профиля при разработке методов, средств и технологий применения объектов профессиональной деятельности.

ПК 1.3. Производить модификацию отдельных модулей информационной системы в соответствии с рабочим заданием, документировать произведенные изменения.

ПК 2.2. Программировать в соответствии с требованиями технического задания.

ПК 2.3. Применять методики тестирования разрабатываемых приложений.

В результате освоения дисциплины студент должен:

иметь представление:

- об основных понятиях и методах, используемых в алгоритмизации;
- о многообразии окружающих нас задач и алгоритмах их решения;
- об особенностях применения алгоритмов и программ при решении прикладных и теоретических задач;

уметь:

- использовать языки программирования, строить логически правильные и эффективные программы;

знать:

- общие принципы построения алгоритмов, основные алгоритмические конструкции;
- понятие системы программирования;
- основные элементы процедурного языка программирования, структуру программы, операции, управляющие структуры, структуры данных, файлы, кассы памяти;
- подпрограммы, составление библиотек программ;
- объектно-ориентированную модель программирования, понятие классов и объектов, их свойств и методов.

4. Структура и содержание дисциплины

4.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Максимальная учебная нагрузка (всего)	153
Обязательная аудиторная учебная нагрузка (всего)	102
в том числе:	
лекции (Л)	60
практические занятия (ПЗ)	42
Самостоятельная работа обучающегося (СР) (всего)	45
в том числе:	
<i>Реферат (Р)</i>	37
<i>Самостоятельное изучение разделов</i>	8
Консультации (К)	6
<i>Итоговая аттестация в форме дифференцированного зачета</i>	

4.2. Тематический план изучения дисциплины

№ раздела	Наименование разделов и их содержание	Количество часов				
		Всего	Аудиторная работа			Вне-ауд. работа СР
			Л	ПЗ	К	
1	Основы алгоритмизации	2			2	
1.1	Понятие алгоритма. Свойства алгоритма. Способы описания	3	2	-		1
1.2	Базовые структуры алгоритмов (развилка, цикл и следование).	3	2	-		1
1.3	Составление словесного и графического алгоритма. Решение задач.	5	2	2		1
	Программирование	4			4	
2.1	Структура программы в Turbo Pascal. Исполнимая и декларативная части. Операторы ввода вывода.	5	2	2		1
2.2	Составление алгоритмов и программ линейной структуры. Решение задач.	5	2	1		2
2.3	Разветвляющиеся структуры в Turbo Pascal. Полная и неполная условные конструкции. Оператор выбора.	4	2	1		1
2.4	Составление алгоритмов и программ на разветвляющиеся структуры.	4	2	1		1
2.5	Циклические структуры в Turbo Pascal. Цикл с предусловием, с постусловием, с параметром.	5	2	1		2
2.6	Составление алгоритмов и программ на циклические структуры.	6	2	2		2
2.7	Процедуры и функции в Turbo Pascal. Взаимодействие параметров.	6	2	2		2
2.8	Составление алгоритмов и программ с использованием процедур и функций.	5	2	2		1
2.9	Одномерные массивы. Способы заполнения массива элементами.	6	2	2		2

2.10	Составление алгоритмов и программ на одномерные массивы. Решение задач.	4	2	-		2
2.11	Двумерные массивы.	6	2	2		2
2.12	Комбинированные типы данных. Записи	3	2	-		1
2.14	Составление программы по теме Записи	5	2	2		1
2.15	Оператор With. Вложенные(иерархические записи)	6	2	2		2
2.16	Строки. Описание строкового типа. Строковые выражения.	6	2	2		2
2.17	Действия со строками (операции объединения, отношения)	6	2	2		2
2.18	Строковые процедуры и функции	6	2	2		2
2.19	Файлы. Понятия логического и физического файла.	4	2	-		2
2.20	Назначение, открытие и закрытие файла.	4	2			2
2.21	Составление программы по теме: "Простой текстовый файл"	4	2	2		
2.22	Общие средства для работы с файлами. Процедуры и функции Модуля System	6	2	2		2
2.23	Модуль DOS. Текстовый файл.	5	2	1		2
2.24	Типизированные файлы. Нетипизированные файлы.	4	2	1		1
2.25	Составление программы по теме: "Типизированный файл"	5	2	2		1
2.26	Составление программы по теме: "Нетипизированные файлы."	5	2	2		1
2.27	Объектно-ориентированное программирование. Основные понятия	1	-	-	-	1
2.28	Соотношение основных понятий. Характеристические свойства основных понятий: наследование, полиморфизм, инкапсуляция	3	2	-	-	1
2.29	Анализ ООП. Наследование записей. Экземпляр типа Object.	5	2	2		1
	Итого:	153	60	42	6	45

4.3. Содержание разделов дисциплины

№ разделов	Наименование разделов	Содержание разделов	Форма текущего контроля
1	2	3	4
1	Основы алгоритмизации		
1.1	Понятие алгоритма	Свойства алгоритма. Способы описания	Лабораторные работы, собеседование
1.2	Базовые структуры алгоритмов	Базовые структуры алгоритмов (развилка, цикл и следование). Составление словесного и графического алгоритма. Решение задач.	собеседование
1.3			самостоятельная работа
2	Программирование		

2.1	Структура программы в Turbo Pascal.	Исполнимая и декларативная части. Операторы ввода вывода.	Самостоятельная работа, тест
2.2	Разветвляющиеся структуры в Turbo Pascal.	Полная и неполная условные конструкции. Оператор выбора. Составление алгоритмов и программ на разветвляющиеся структуры.	Лабораторные работы, тест, самостоятельная работа
2.3	Циклические структуры в Turbo Pascal.	Цикл с предусловием, с постусловием, с параметром.	Лабораторные работы, самостоятельная работа
2.4	Процедуры и функции в Turbo Pascal.	Взаимодействие параметров. Составление алгоритмов и программ с использованием процедур и функций	Лабораторные работы Опрос, самостоятельная работа
2.5	Одномерные массивы. Способы заполнения массива элементами.	Составление алгоритмов и программ на одномерные массивы. Формирование одномерного массива -Нахождение суммы элементов -Нахождение элементов с заданным свойством	Лабораторные работы , опрос, самостоятельная работа
2.6	Двумерные массивы.	Формирование двумерного массива -Удаление и вставка строк или столбцов матрицы	Лабораторные работы , Самостоятельная работа
2.7	Комбинированные типы данных. Записи	Составление программы по теме Записи. Оператор With. Вложенные(иерархические записи). Записи с вариантами.	Лабораторные работы , собеседование, решение задач
2.8	Строки. Описание строкового типа. Строковые выражения.	Действия со строками (операции объединения, отношения) Строковые процедуры и функции	Лабораторные работы Диктант. Решение задач. Зачетный урок
2.9	Файлы. Понятия логического и физического файла.	Назначение, открытие и закрытие файла. Модуль DOS. Текстовый файл. Составление программы по теме: "Простой текстовый файл"	Лабораторные работы, собеседование, решение задач
2.10	Общие средства для работы с файлами.	Процедуры и функции Модуля System, DOS	Лабораторные работы Диктант.
2.11	Типизированные файлы.	Составление программы по теме: "Типизированный файл".	Лабораторные работы
2.12	Нетипизированные файлы.	Составление программы по теме: "Нетипизированные файлы."	Лабораторные работы
3.Объектно-ориентированное программирование.			
3.1	Основные понятия	Соотношение основных понятий Характеристические свойства: наследование, полиморфизм, инкапсуляция	Лабораторные работы собеседование
3.2	Наследование записей	Анализ ООП. Наследование записей. Экземпляр типа Object.	Зачетное занятие по основным понятиям

4.4. Лабораторные работы

№ ЛР	Наименование лабораторных работ	Кол-во часов
1	Составление словесного и графического алгоритма индивидуального задания	2
2	Среда Турбо-Паскаль. Главное меню. Работа с окнами	2
3	Составление программ линейной структуры	2
4	Составление программ на разветвляющиеся структуры.	2
5	Составление программ символьные переменные в разветвляющейся структуре	2
6	Составление программ на циклические структуры	2
7	Составление программ с использованием процедур и функций	2
8	Выполнение индивидуальной работы по темам: циклические структуры, процедуры и функции	2
9	Выполнение индивидуальной работы по теме: одномерные массивы.	2
10	Составление программ на одномерные и двумерные массивы	2
11	Составление программы по теме: Записи. Выполнение индивидуального задания	4
12	Составление программы по теме: "строки" действия со строками (операции объединения, отношения) отладка и редактирование	2
13	Выполнение индивидуального задания. Строковые процедуры и функции	2
14	Создание и отладка программы "Простой текстовый файл"	2
15	Процедуры и функции модуля System и модуля DOS.	4
16	Типизированный, нетипизированный файл. Составление и отладка программы по индивидуальному заданию	4
17	«Простейший объект». Поля объекта. Определение метода инициализации полей объекта.	2
18	Выполнение индивидуального задания. Составление программы наследование записи. Экземпляр типа Object.	2
Всего:		42

4.5. Самостоятельное изучение разделов дисциплины

№ Раздела	Вопросы, выносимые на самостоятельное изучение	Кол-во часов
1	Этапы решения задач на ПК. Составление технического задания. Рабочая документация. Ввод в действие.	2
1	Основные понятия структур данных. Примеры простых программ	2
1	Арифметика Паскаля. Типы данных. Порядковые типы данных. Логика Паскаля. Сложные условия. Тип Boolean. Совместимость типов	2
2	Принципы представления программ	4
2	Примеры построения комбинированных программ обработки массивов.	5
2	Применение рекурсии для обработки структур данных	4
3	Алгоритмы и программы обработки текстов	6
8	Множества и математика. Примеры построения программ обработки множества	4
9	Обработка записи. Последовательный поиск	4
10	Общие свойства файлов. Примеры программ обработки файлов	4

11	Модульное программирование.	4
12	Объекты и модули. Объекты изнутри. Взаимодействие объектов.	4
Итого:		45

5. Образовательные технологии

Выбор организационной формы работы, соответствующей типу выполняемого задания, а также эффективное руководство и управление деятельностью студентов, ее регулирование на занятии способствует интенсификации процесса обучения.

В процессе преподавания данной дисциплины используются как классические методы обучения (лекции, традиционные лабораторные работы), так и различные виды самостоятельной работы студентов по заданию преподавателя, которые направлены на развитие творческих качеств студентов и на поощрение их интеллектуальных инициатив.

5.1 Интерактивные образовательные технологии, используемые в аудиторных занятиях

Вид занятия (Л, ПР)	Используемые интерактивные образовательные технологии	Кол-во часов
Л ПР	Пассивный метод: опрос, самостоятельная работа, тест. Активный и интерактивный методы: - Работа в команде при выполнении конкретного задания как во время аудиторного занятия, так и в рамках самостоятельной работы; - Метод «дебаты»; - Работа над презентационными проектами по курсу Изучение и закрепление нового материала: работа с видео материалом.	20
Итого:		20

6. Оценочные средства для текущего контроля успеваемости и промежуточной аттестации

В процессе оценки знаний студентов по программам ФГОС ВПО необходимо использовать как традиционные, так и инновационные типы, виды и формы контроля.

В разделе указывается:

- Перечень рекомендованных вопросов к контрольным работам;
- Тестовые проверочные задания текущего контроля и промежуточной аттестации
- Тематика рефератов с методическими материалами;
- Вопросы к зачету по дисциплине;

Технические формы контроля

Данные формы контроля осуществляются с привлечением разнообразных технических средств и могут содержать:

- программы компьютерного тестирования
- учебные задачи
- комплексные ситуационные задания

- электронный практикум содержит набор заданий, которые необходимо выполнить студенту.

Предъявляемое задание выбирается из базы данных и закрепляется за конкретным студентом. В отличие от тестов, задание, которое предъявляется студенту в рамках практикума, не требует мгновенного выполнения. Системой определяется срок, в течение которого задание должно быть сдано. Результатом выполнения задания должен быть файл, отсылаемый студентом в базу данных

6.1 Образец тестового и зачетного задания

Вариант 1

1. Какой из перечисленных разделов обязателен в программе:
 - а) раздел var?
 - б) раздел const?
 - в) раздел begin...end?
 - г) раздел Type?
2. Укажите тип переменной, которая будет содержать год основания города:
 - а) Real; б) Integer; в) Char; г) String.
3. Какой из перечисленных операторов является оператором цикла с предварительным условием:
 - а) Repeat...Until...;
 - б) for...to...do...;
 - в) While...do...;
 - г) if...then...else....
4. Укажите правильную форму записи процедуры:
 - а) Procedure <имя>(<фактические параметры>:тип);
 - б) Procedure <имя>(<формальные параметры>:тип);
 - в) Procedure <имя>(<фактические параметры>):<тип результата>;
 - г) Procedure <имя>(<формальные параметры>):<тип результата>.
5. Какое значение будет принимать переменная Y после выполнения фрагмента программы:


```
Var y,x:integer;
  Begin  y:=0; for x:=1 to 9 do y:=y+x; writeln('y=',y);  end.
```

 - а) 45 б) 0 в) 9 г) 10
6. Как осуществляется обращение к отдельному элементу одномерного массива:
 - а) <имя массива>:=<индекс элемента>;
 - б) <имя массива>(<индекс элемента>);
 - в) <имя массива>[<индекс элемента>];
 - г) <имя массива>[<элемент>].
7. Какая строковая процедура вставляет подстроку в строку:
 - а) Str; б) Val; в) Insert; г) Copy.
8. Ограниченный, неупорядоченный набор различных элементов одного типа называется:
 - а) массив; б) запись; в) строка; г) множество.
9. Структурированный тип данных, состоящий из фиксированного числа компонентов одного или нескольких типов это:
 - а) Массив; б) Файл; в) Запись; г) Строковый;
10. Создать подкаталог:

а) Chdir ('Номер 1'); б) Mkdir ('Номер 1'); в) Rmdir ('Номер 1'); г) Nmdir('Номер 1');

11. Для работы с физическим файлом необходимо связать его с:

- а) файловой константой;
- б) файловой переменной;
- в) файловой частотой;
- г) файловой касательной

12. Вставка строки str1 в строку str2 начиная с позиции Poz:

- а) Concat (str1, str2, ..., strN); б) Copy (st, Poz, N); в) Insert (str1, str2, Poz);
- г) Delete (str1, str2, Poz);

13. Точная формулировка решения задачи на ПК с описанием входной и выходной информации это -...

- а) алгоритм
- б) постановка задачи
- в) программа
- г) предметная область

6.2 Контрольные вопросы для самопроверки

Раздел 1. Структура программы в Турбо Паскале.

- 1) Структура программы в Турбо Паскале.
- 2) Какие слова являются зарезервированными в Турбо Паскале?
- 3) Главное меню.
- 4) Каковы возможности и в чем преимущества интегрированной среды программирования?
- 5) Перечислите основные файлы среды программирования Турбо Паскаль и их назначение. Как запустить среду программирования Турбо Паскаль?
- 6) Перечислите основные компоненты окна редактирования программ среды программирования Турбо Паскаль. В чем их назначение?
- 7) Опишите значение информации в строке статуса окна редактирования интегрированной среды программирования:
(F1 Help; F2 Save; F3 Open; Alt+F9 Compile; F9 Make; Alt+F10 Local menu)
- 8) Как откомпилировать файл программы?
- 9) В чем отличие пункта Run от пункта Compile главного меню интегрированной среды программирования?
- 10) Как посмотреть результаты выполнения программы в окне пользователя?

Раздел 2. Составление программ линейной структуры.

- 1) Что такое переменная?
- 2) В каком порядке выполняются строки программы?
- 3) Что такое программа линейной структуры?
- 4) Типы данных в Turbo Pascal.
- 5) Интерпретация блоков при графическом способе записи алгоритма.

Раздел 3. Разветвляющие структуры. Поливариантные условия.

- 1) Что такое оператор? Чем отличаются простые и структурные операторы?
- 2) Оператор безусловного перехода, его назначение и особенности применения.

3) Что представляет собой составной оператор? Как ограничиваются операторы, объединенные в составной оператор?

4) Назначение, формы записи и порядок выполнения оператора условия if.

5) Особенности использования вложенных условных операторов.

6) Каковы отличия оператора выбора case от оператора условия if?

7) Какие правила должны выполняться при использовании оператора выбора case?

Раздел 4. Операторы цикла.

1) Что такое Циклы?

2) Какая базовая структура соответствует циклу “До” и “Пока”?

3) Формат строки цикла с параметрами?

4) Когда используется оператор цикла с параметрами?

5) Формат строки цикла с предусловием?

6) Формат строки цикла с послесловием?

Раздел 5. Процедуры и функции. Взаимодействие параметров.

1) Что называется подпрограммой? В чем состоит сходство и различие подпрограмм- процедур и подпрограмм-функций в языке Турбо Паскаль?

2) В чем различие между стандартными и определенными пользователем подпрограммами? Приведите примеры.

3) Запишите синтаксическую диаграмму определения процедуры, функции.

4) Опишите последовательность событий при вызове процедуры или функции.

5) Что называется параметром, и каково его назначение? Формальные, фактические параметры, их взаимосвязь.

6) Каковы отличия параметров-значений от параметров-переменных, особенности их описания и применения.

7) Каковы особенности параметров-процедур и параметров-функций?

8) Чем отличаются локальные и глобальные параметры? Какова область их действия?

9) В каких случаях требуется предварительное или внешнее описание подпрограмм? Каковы особенности использования подпрограмм с предварительным описанием?

Раздел 6. Структурированные типы данных. Массивы

1) Что такое массив?

2) Как определить местоположение элемента в массиве?

3) Что такое индекс?

4) Особенности расположения элементов массива в памяти ЭВМ?

5) Как объявить в программе массив (одномерный, двумерный).

6) Общие и отличительные черты одномерных, двумерных и n – мерных массивов?

7) Какие действия можно производить над массивом целиком и при каких условиях?

8) Как обратиться к элементу массива?

9) Какие операции можно производить над элементами массива?

10) Какими способами можно ввести (вывести) элементы массива?

11) Почему при описании массива предпочтительнее употреблять константы, а не указывать размеры массива в явном виде?

12) Какие способы сортировки массивов вы знаете?

Раздел 6.1 Структурированные типы данных. Строки.

- 1) Что такое строка?
- 2) Каким идентификатором определяются данные строкового типа?
- 3) Какова максимально возможная длина строки? Как определить текущую длину строки?
- 4) Какие выражения называют строковыми?
- 5) Какие операции допустимы над строковыми данными?
- 6) Каким образом производится сравнение строк?
- 7) Как можно обратиться к отдельным символам строки?
- 8) Как производится сцепление строк?
- 9) Назначение специальных процедур и функций обработки данных строкового типа. Привести примеры.

Раздел 6.2 Структурированные типы данных. Записи.

- 1) Что такое запись?
- 2) Отличие записи от массива?
- 3) Что из себя представляют элементы записи?
- 4) Формат записи?
- 5) Для чего нужен оператор With?

Раздел 6.3 Структурированные типы данных. Файлы.

- 1) Что такое файл ?
- 2) Для каких целей применяются файлы ?
- 3) Характерные особенности файлов ?
- 4) Какие устройства компьютера Т.Р. использует наряду с файлами ? Каковы логические имена этих устройств ?
- 5) Затем используется специальная файловая переменная ?
- 6) Какие процедуры служат для открытия файлов ? Их общие черты и отличие?
- 7) Затем применяется процедура Close ?
- 8) Для каких целей используется специальная функция IOResult ?
- 9) Дать определение текстового файла ? Из чего состоит содержимое текстового файла ?
- 10) Назначение процедуры Append Назначение и отличительные особенности процедур Read и Write от Readln и Writeln.

Раздел 7. Усложненные структуры программ. Модули.

- 1) Какие файлы сохраняются на диске после трансляции программы;
- 2) Что позволяет выполнить встроенный в ТР отладчик;
- 3) Перечислить способы выполнения программы до определенной точки.

Раздел 8. Объектно-ориентированное программирование. Основные понятия

- 1) Что такое ООП?
- 2) Какая основная цель ОО-методологии?
- 3) В чем состоит отличие типа «Объект» от типа «Запись».
- 4) Что называется объектом, свойством, методом, классом?
- 5) Что такое инкапсуляция?

6.3 Перечень вопросов для экзамена

1. Язык программирования Turbo Pascal. Назначение. Причины использования языка.
 2. Использование среды программирования Turbo Pascal.
 3. Структура программы в Turbo Pascal.
 4. Арифметика и алфавит Turbo Pascal. Типы данных.
 5. Преобразование типов и действия над ними. Константы и переменные.
 6. Декларативная и исполнимая части программы в Turbo Pascal.
 7. Организация программ линейной структуры.
 8. Процедуры ввода – вывода. Особенности использования.
 9. Организация программ разветвляющей структуры.
 10. Условная конструкция (виды, особенности использования).
 11. Оператор выбора. Структура записи. Особенности использования.
 12. Организация программ циклической структуры.
 13. Оператор цикла с предусловием. Структура записи. Особенности использования.
 14. Оператор цикла с постусловием. Структура записи. Особенности использования.
 15. Оператор цикла со счётчиком (с параметром). Структура записи. Особенности использования.
 16. Сравнение работы операторов повторения. Вложенные циклы.
 17. Процедуры и функции в Turbo Pascal. Определения. Особенности.
 18. Процедуры и функции пользователя. Форматы описаний. Формальные и фактические параметры.
 19. Виды параметров при описании подпрограмм. Назначение. Особенности использования.
 20. Особенности оформления подпрограммы в виде процедуры.
 21. Особенности оформления подпрограммы в виде функции.
 22. Понятие рекурсии. Рекурсивные процедуры и функции.
 23. Строки. Описание строкового типа данных. Строковые выражения.
 24. Операции над строковыми типами данных (объединение, отношение).
 25. Подпрограммы для работы со строками (Pos, Concat).
 26. Подпрограммы для работы со строками (Val, Delete).
 27. Подпрограммы для работы со строками (Length, Copy).
 28. Подпрограммы для работы со строками (Insert, Str).
 29. Данные типа «массив». Одномерный массив. Определение. Формат описания.
- Примеры.
30. Двумерные массивы. Определение. Формат описания. Примеры.
 31. Действия над массивами. Особенность расположения в памяти. Пример описания.
 32. Действия над элементами массива. Способы заполнения массива элементами.
 33. Понятие сортировки. Виды сортировок.
 34. Множественный тип данных. Определение. Формат описания.
 35. Операции над множествами (объединение, пересечение, сравнение).опрос
завтра
 36. Данные типа «запись». Определение. Формат описания. Пример.
 37. Данные типа «запись». Обращение к полям. Два способа обращения.

38. Составное имя в записях. Оператор With (назначение, формат записи).
39. Иерархические (вложенные) записи. Пример.
40. Применимые операции над полями записи и над записью в целом.
41. Файловый тип данных в Turbo Pascal. Необходимость применения файлового типа.
42. Понятие физического и логического файла. Целесообразность применения файлового типа.
43. Классификация файлов в Turbo Pascal.
44. Файловая переменная. Процедура Assign.
45. Процедуры открытия и закрытия файлов. Особенности использования.
46. Назначение функций IOResult, Eof, Eoln.
47. Назначение процедур Rename, Erase, Seek, Truncate.
48. Процедуры работы с каталогами. Назначение, формат описания.
49. Текстовые файлы. Процедура Append.
50. Типизированные файлы. Особенности использования.
51. Нетипизированные файлы. Особенности использования.
52. Модульное программирование Свойства программного модуля. Функционально модульная структура приложения. (Привести пример Функционально-модульной структуры приложения вашего индивидуального задания).
53. «Объектно-ориентированное программирование. Основные понятия. Соотношение основных понятий»: Что такое ООП? Что называется объектом, свойством, методом, событием, классом?
54. Характеристические свойства основных понятий: наследование, полиморфизм, инкапсуляция. Анализ ООП
55. Наследование записей. Экземпляр типа Object. Правила наследования.
56. Что общего имеет тип «Объект» с типом «Запись»? В чем состоит отличие типа «Объект» от типа «Запись»?

6.4. Темы рефератов для самостоятельной работы

1. Первая программист Августа Ада Лавлейс
2. История программирования (эволюция программирования)
3. Роль информационных технологий сегодня
4. Архитектура и возможности семейств языков высокого уровня
5. Экономические и правовые аспекты информационных технологий
6. Программирование рекурсивных алгоритмов
7. Объектно-ориентированное программирование. Достоинства и недостатки.

6.5 Методические рекомендации студентам по организации изучения дисциплины

Успешное освоение курса предполагает активное, творческое участие студента на всех и этапах ее освоения путем планомерной, повседневной работы.

Общие рекомендации

Изучение дисциплины следует начинать с проработки настоящей рабочей программы, особое внимание, уделяя целям, задачам, структуре и содержанию курса.

Работа с конспектом лекций

Просмотрите конспект сразу после занятий. Пометьте материал конспекта лекций, который вызывает затруднения для понимания. Попытайтесь найти ответы на

затруднительные вопросы, используя предлагаемую литературу. Если самостоятельно не удалось разобраться в материале, сформулируйте вопросы и обратитесь на текущей консультации или на ближайшей лекции за помощью к преподавателю.

Каждую неделю рекомендуется отводить время для повторения пройденного материала, проверяя свои знания и умения и навыки по контрольным вопросам.

Выполнение лабораторных работ

На первом занятии получите у преподавателя график выполнения лабораторно-практических работ на текущий семестр. Обзаведитесь всем необходимым методическим обеспечением.

Перед посещением лабораторно-практических изучите теорию вопроса, предполагаемого к исследованию, ознакомьтесь с руководством по соответствующей работе.

Оформление отчетов должно производиться после окончания работы непосредственно в аудитории, при наличии свободного времени или дома.

Для подготовки к защите отчета следует проанализировать разработанные программные средства, сопоставить их с целевым назначением задания, обобщить результаты исследований в виде лаконичных выводов по работе.

6.6. Методические рекомендации преподавателям по методике проведения основных видов учебных занятий.

Лекции. Методика чтения лекций

Лекции являются одним из основных методов обучения по дисциплине «Теория алгоритмов», которые должны решать следующие задачи:

- изложить важнейший материал программы курса;
- развить у студентов потребность к самостоятельной работе над учебником и научной литературой;

Главной задачей каждой лекции является показ сущности темы и анализ ее основных положений. Рекомендуется на первой лекции довести до студентов структуру курса и его разделы, а в дальнейшем указывать начало каждого раздела, суть и его задачи, а, закончив изложение подводить итог по этому разделу, чтобы связать его со следующим.

Содержание лекций

Содержание лекций определяется настоящей рабочей программой курса. Крайне желательно, чтобы каждая лекция охватывала и исчерпывала определенную тему курса и представляла собой логически вполне законченную работу. Лучше сократить тему, но не допускать перерыва ее на таком месте, когда основная идея еще полностью не раскрыта.

Методика проведения практических занятий

Целями проведения практических работ являются:

- установление связей теории с практикой в форме экспериментального подтверждения положений теории;
- установлению степени соответствия современных требований к разработке ПО с полученными результатами;
- обучение студентов умению анализировать полученные результаты, сопоставлять их с теоретическими положениями;
- контроль самостоятельной работы студентов по освоению курса;

Цели практикума достигаются наилучшим образом в том случае, если выполнение эксперимента предшествует определенной подготовительная внеаудиторная работа.

Поэтому преподаватель обязан довести до всех студентов график выполнения практических работ на весь семестр с тем, чтобы они могли заниматься целенаправленной домашней подготовкой.

Перед началом очередного занятия преподаватель должен удостовериться в готовности студентов к выполнению очередной работы путем короткого собеседования и проверки наличия у студентов заготовленных протоколов проведения работы.

Проведение лабораторных работ осуществляется в компьютерных классах (аудитории 202,221-226,105,).

7. Учебно-методическое обеспечение дисциплины

7.1. Основная литература

1. Рапаков, Г.Г. Turbo Pascal для студентов и школьников /Г.Г.Рапаков, С.Ю.Ржеуцкая. - СПб.: БХВ-Петербург, 2011.-252 с.

2. Семакин И.Г. Основы алгоритмизации и программирования: уч. пособие для СПО /И.Г.Семакин, А.П.Шестаков. – М.: Академия, 2011. – 400 с.

7.2. Дополнительная литература

1. Фаронов В.В. TURBO PASCAL 7.0 Практика программирования: учеб. пособие /В.В.Фаронов. –М.: КНОРУС, 2012. -414 с.

7.3. Интернет-ресурсы

1. Дистанционное обучение [Электронный ресурс]: портал. – Режим дос-тупа: <http://do.rksi.ru/site/help/help8.xml>

2. Паскаль (язык программирования): материал из Википедии — свободной энциклопедии [Электронный ресурс]: сайт. – Режим доступа: [https://ru.wikipedia.org/wiki/Паскаль_\(язык_программирования\)](https://ru.wikipedia.org/wiki/Паскаль_(язык_программирования))

3. Информационный портал. Все о компьютере и программировании для начинающих info-comp.ru - Здесь есть все! [Электронный ресурс]- Режим доступа:

- <http://info-comp.ru/programmirovanie/67-turbopascal-.html>.

7.4. Программное обеспечение современных информационно-коммуникационных технологий

Для изучения данной дисциплины необходим целый комплекс технических средств, использующийся как основной элемент для усвоения практического материала, умения использовать технические средства в работе. Необходимым и обязательным средством является персональный компьютер современной конфигурации при наличии современного программного компьютерного обеспечения – Windows Me, Microsoft Word,– версии не ниже 2003 года, Borland Pascal, PascalABC.

8. Материально-техническое обеспечение дисциплины

Для изучения дисциплины необходимо следующее материально-техническое обеспечение:

- аудитории;
- библиотечный фонд;
- научный фонд;
- интернет-аудитории;
- компьютерная аудитория.