

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
(ВлГУ)

УТВЕРЖДАЮ

Проректор
по образовательной деятельности

А.А. Панфилов

« 28 » 08 2018 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
ПРОГРАММИРОВАНИЕ**

Направление подготовки 44.03.05 – Педагогическое образование

Профиль/программа подготовки Математика. Информатика

Уровень высшего образования бакалавриат

Форма обучения очная

Семестр	Трудоемкость зач. ед./ час.	Лекции, час.	Практич. занятия, час.	Лаборат. работы, час.	СРС, час.	Форма промежуточной аттестации (экзамен/зачет/зачет с оценкой)
1	2/72	18	18	18	18	зачет
2	3/108	18	18	18	18	Экзамен 36
3	5/180	18	18	18	81	Экзамен 45
Итого	10/360	54	54	54	117	Зачет, экзамен 81

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель освоения дисциплины: развитие логического и алгоритмического мышления у студентов.

Задачи:

- формирование и развитие умений планирования действий и структурирования информации, необходимых для решения поставленной задачи, при помощи фиксированного набора средств;
- изучение формального языка как средства управления компьютером;
- формирование способности отображать реальные объекты и явления в компьютерные информационные структуры;
- использование языка программирования как средства автоматизации вычислений, необходимых в учебной деятельности.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина «Программирование» относится к вариативной части учебного плана 44.03.05 – Педагогическое образование.

Пререквизиты дисциплины: дисциплина опирается на знания предметов основной образовательной программы среднего образования: «Информатика и ИКТ» (содержательная линия «Алгоритмизация и программирование»). Материал, изучаемый в ходе освоения дисциплины «Программирование» является базовым для дальнейшей работы с языками программирования.

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП

Код формируемых компетенций	Уровень освоения компетенции	Планируемые результаты обучения по дисциплине характеризующие этапы формирования компетенций (показатели освоения компетенции)
1	2	3
ОК-6	Частичное освоение компетенции	<i>Знать:</i> <ul style="list-style-type: none">- этапы решения задач с использованием компьютера;- этапы выполнения компьютером программ, написанных на языке высокого уровня. <i>Уметь:</i> <ul style="list-style-type: none">- организовывать действия созданного алгоритма с использованием выбранного языка программирования;- реализовывать ручное исполнение алгоритма. <i>Владеть:</i> <ul style="list-style-type: none">- навыками проектирования программ на языке программирования высокого уровня.
ПК-1	Частичное освоение компетенции	<i>Знать:</i> <ul style="list-style-type: none">- понятие алгоритма, его свойства, способы записи, исполнителей алгоритмов в объеме, не меньшем, чем данный раздел представлен в программе школьного курса «Информатики и ИКТ»;- язык программирования как исполнитель алгоритма;- назначение систем программирования;- этапы решения задач с использованием компьютера;- этапы выполнения компьютером программ, написанных на языке. <i>Уметь:</i> <ul style="list-style-type: none">- составлять и записывать в различных формах алгоритмы решения практических и прикладных задач;- анализировать разработанный алгоритм с точки зрения его работоспособности. <i>Владеть:</i> <ul style="list-style-type: none">- основными методами проектирования и отладки алгоритмов.

ПК-11	Частичное освоение компетенции	<p><i>Знать:</i></p> <ul style="list-style-type: none"> - основные понятия модульного и объектно-ориентированного программирования; - методы построения и отладки программ. <p><i>Уметь:</i></p> <ul style="list-style-type: none"> - организовывать необходимую для решения задач информацию средствами выбранного языка программирования; - разрабатывать внешний интерфейс реализации алгоритма решения задачи в выбранной среде программирования; - анализировать структуру программы с целью ее дальнейшей оптимизации; - оценивать эффективность работы программы.
-------	--------------------------------	--

4. ОБЪЕМ И СТРУКТУРА ДИСЦИПЛИНЫ

Трудоемкость дисциплины составляет 10 зачетных единиц, 360 часов

№ п/п	Наименование тем и/или разделов/тем дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Объем учебной работы, с применением интерактивных методов (в часах / %)	Формы текущего контроля успеваемости, форма промежуточной аттестации (по семестрам)
				Лекции	Практические занятия	Лабораторные работы	СРС		
1	Понятие алгоритма.	1	1-2	2	2	2	2	2/33%	
2	Структура программы.	1	3-4	2	2	2	2	2/33%	
3	Команды ветвления.	1	5-8	4	4	4	4	4/33%	Рейтинг-контроль №1
4	Циклические операторы.	1	9-14	6	6	6	6	6/33%	Рейтинг-контроль №2
5	Структурированные типы данных. Массивы.	1	14-18	4	4	4	4	4/33%	Рейтинг-контроль №3
Всего за 1 семестр				18	18	18	18	18/33%	Зачет
1	Сортировки массивов.	2	1-6	6	6	6	6	6/33%	Рейтинг-контроль №1
2	Символы и строки.	2	7-12	6	6	6	6	6/33%	Рейтинг-контроль №2
3	Подпрограммы.	2	13-18	6	6	6	6	6/33%	Рейтинг-контроль №3
Всего за 2 семестр				18	18	18	18	18/33%	Экзамен 36
1	Структурированные типы данных: запись и множество.	3	1-8	8	8	8	31	8/33%	Рейтинг-контроль №1
2	Работа с файлами.	3	9-12	4	4	4	20	4/33%	Рейтинг-контроль №2
3	Графические возможности среды программирования.	3	13-18	6	6	6	30	6/33%	Рейтинг-контроль №3
Всего за 3 семестр:				18	18	18	81	18/33%	Экзамен 45
Наличие в дисциплине КП/КР					-				
Итого по дисциплине				54	54	54	117	54/33%	Зачет, Экзамен 81

Содержание лекционных занятий по дисциплине

Раздел I. Понятие алгоритма.

Тема 1. Понятие алгоритма. Свойства алгоритма.

Определения и свойства алгоритма. Примеры алгоритмов для неформальных и формальных исполнителей.

Тема 2. Способы записи алгоритма.

Различные способы записи алгоритма: словесный, графический (блок-схемы), псевдо-код, программный. Формальное исполнение алгоритма.

Раздел II. Структура программы.

Тема 3. Структура программы в языке программирования Pascal. Типы данных.

Структура и разделы программы. Алфавит языка. Типы данных. Встроенные функции языка.

Тема 4. Операторы ввода, вывода и присваивания.

Арифметические выражения. Оператор присваивания. Оператор ввода, оператор вывода.

Линейные программы.

Раздел III. Команды ветвления.

Тема 5. Логический тип и логические выражения.

Оператор безусловного перехода. Условный оператор. Логические выражения, их истинность и ложность.

Тема 6. Команда ветвления в полной и неполной форме.

Оператор ветвления в неполной и в полной форме. Примеры программ.

Тема 7. Оператор выбора.

Оператор выбора. Примеры задач. Организация меню.

Раздел IV. Циклические операторы.

Тема 8. Команды цикла с предусловием и постусловием.

Цикл с предусловием. Цикл с постусловием. Простые и составные условия.

Тема 9. Циклы с параметром.

Цикл с параметром. Примеры программ. Преобразование циклов из одного вида в другой.

Раздел V. Структурированные типы данных. Массивы.

Тема 10. Линейные массивы.

Способы описания массива. Заполнение массива. Случайные числа. Ввод и вывод элементов массива.

Раздел VI. Сортировки массивов.

Тема 11. Алгоритмы поиска и сортировок в линейном массиве.

Сортировки «простой выбор», «простой обмен», «простые вставки». Алгоритм простого перебора. Метод бинарного поиска.

Тема 12. Двумерные массивы.

Двумерные массивы. Способы описания массива. Заполнение массива случайным образом и с клавиатуры. Ввод и вывод элементов массива.

Раздел VII. Символы и строки.

Тема 13. Функции для работы с символьным типом.

Символьный тип. Функции для работы с символьным типом. Примеры и задачи.

Тема 14. Строковые величины. Функции для работы со строковыми.

Строковый тип. Операции над строковыми данными. Примеры и задачи.

Раздел VIII. Подпрограммы.

Тема 15. Подпрограмма-процедура и подпрограмма-функция.

Понятие подпрограммы. Отличие процедуры от функции. Вызов процедуры и функции.

Рекурсия.

Тема 16. Параметры процедур и функций.

Формальные и фактические параметры. Передача параметра по ссылке и по значению. Вызов процедуры и функции. Рекурсия.

Тема 17. Рекурсивные подпрограммы.

Понятие рекурсии. Примеры задач с использованием рекурсии. Ручное исполнение программы с рекурсией.

Раздел IX. Структурированные типы данных: запись и множество.

Тема 18. Структурированный тип данных записи.

Тип Запись. Операции над записями. Оператор присоединения. Примеры использования типа запись.

Тема 19. Массивы записей.

Массивы записей. Примеры программ с использованием массивов записей.

Тема 20. Тип множество.

Тип множество. Построение множества. Действия над множествами. Вывод множества на экран. Примеры решения задач с использованием типа множество.

Раздел X. Работа с файлами.

Тема 21. Чтение и запись информации из файлов.

Файловые переменные. Способы открытия файла (чтение, запись, дозапись). Чтение и запись из файла.

Тема 22. Типизированные и текстовые файлы. Операции для работы с файлами.

Файлы с типом. Текстовые файлы. Организация чтения из текстового файла. Примеры программ.

Раздел XI. Графические возможности среды программирования.

Тема 23. Процедуры и функции модуля CRT.

Модуль CRT. Управление курсором. Управление цветом.

Тема 24. Процедуры и функции модуля Graph.

Процедуры и функции модуля Graph. Графические операторы. Графические примитивы.

Тема 25. Организация анимации средствами среды программирования.

Построение графиков функций. Анимация средствами языка.

Содержание практических занятий по дисциплине

Практическое занятие 1. Операторы ввода-вывода и присваивания.

Арифметические выражения. Оператор присваивания. Оператор ввода, оператор вывода.

Линейные программы.

Практическое занятие 2. Логический тип и логические выражения.

Оператор безусловного перехода. Условный оператор. Логические выражения, их истинность и ложность.

Практическое занятие 3. Команда ветвления в полной и неполной форме.

Оператор ветвления в неполной и в полной форме. Примеры программ.

Практическое занятие 4. Оператор выбора.

Оператор выбора. Примеры задач. Организация меню.

Практическое занятие 5. Циклические операторы. Команды цикла с предусловием и постусловием.

Цикл с предусловием. Цикл с постусловием. Простые и составные условия.

Практическое занятие 6. Циклы с параметром.

Цикл с параметром. Примеры программ. Преобразование циклов из одного вида в другой.

Практическое занятие 7. Линейные массивы.

Способы описания массива. Заполнение массива. Случайные числа. Ввод и вывод элементов массива.

Практическое занятие 8. Сортировки массивов.

Сортировки «простой выбор», «простой обмен», «простые вставки».

Практическое занятие 9. Алгоритмы поиска в линейном массиве.

Алгоритм простого перебора. Метод бинарного поиска.

Практическое занятие 10. Двумерные массивы.

Двумерные массивы. Способы описания массива. Заполнение массива случайным образом и с клавиатуры. Ввод и вывод элементов массива.

Практическое занятие 11. Решение задач на двумерные массивы.

Сумма матриц. Произведение матриц. Вывод на экран матрицы в принятом в математике виде.

Практическое занятие 12. Функции для работы с символьным типом.

Символьный тип. Функции для работы с символьным типом. Примеры и задачи.

Практическое занятие 13. Строковые величины. Функции для работы со строковыми.

Строковый тип. Операции над строковыми данными. Примеры и задачи.

Практическое занятие 14. Алгоритмы шифрования данных.

Строковый тип. Операции над строковыми данными. Примеры и задачи.

Практическое занятие 15. Подпрограмма-процедура.

Понятие подпрограммы. Отличие процедуры от функции. Вызов процедуры и функции.

Практическое занятие 16. Подпрограмма-функция.

Понятие подпрограммы. Отличие процедуры от функции. Вызов процедуры и функции.

Практическое занятие 17. Параметры процедур и функций.

Формальные и фактические параметры. Передача параметра по ссылке и по значению. Вызов процедуры и функции.

Практическое занятие 18. Рекурсивные подпрограммы.

Понятие рекурсии. Примеры задач с использованием рекурсии. Ручное исполнение программы с рекурсией.

Практическое занятие 19. Структурированный тип данных запись.

Тип Запись. Операции над записями. Оператор присоединения. Примеры использования типа запись.

Практические занятия 20-21. Массивы записей.

Массивы записей. Примеры программ с использованием массивов записей.

Практическое занятие 22. Тип множество.

Тип множество. Построение множества. Действия над множествами. Вывод множества на экран. Примеры решения задач с использованием типа множество.

Практическое занятие 23. Работа с файлами. Чтение и запись информации из файлов.

Файловые переменные. Способы открытия файла (чтение, запись, дозапись). Чтение и запись из файла.

Практическое занятие 24. Типизированные и текстовые файлы. Операции для работы с файлами.

Файлы с типом. Текстовые файлы. Организация чтения из текстового файла. Примеры программ.

Практическое занятие 25. Графические возможности среды программирования. Процедуры и функции модуля CRT.

Модуль CRT. Управление курсором. Управление цветом.

Практическое занятие 26. Процедуры и функции модуля Graph.

Процедуры и функции модуля Graph. Графические операторы. Графические примитивы.

Практическое занятие 27. Организация анимации средствами среды программирования.

Построение графиков функций. Анимация средствами языка.

Содержание лабораторных занятий по дисциплине

Лабораторная работа №1. Операторы ввода-вывода и присваивания.

Арифметические выражения. Оператор присваивания. Оператор ввода, оператор вывода.

Линейные программы.

Лабораторная работа № 2. Логический тип и логические выражения.

Оператор безусловного перехода. Условный оператор. Логические выражения, их истинность и ложность.

Лабораторная работа № 3. Команда ветвления в полной и неполной форме.

Оператор ветвления в неполной и в полной форме. Примеры программ.

Лабораторная работа № 4. Оператор выбора.

Оператор выбора. Примеры задач. Организация меню.

Лабораторная работа № 5. Циклические операторы. Команды цикла с предусловием и постусловием.

Цикл с предусловием. Цикл с постусловием. Простые и составные условия.

Лабораторная работа № 6. Циклы с параметром.

Цикл с параметром. Примеры программ. Преобразование циклов из одного вида в другой.

Лабораторная работа № 7. Линейные массивы.

Способы описания массива. Заполнение массива. Случайные числа. Ввод и вывод элементов массива.

Лабораторная работа № 8. Сортировки массивов.

Сортировки «простой выбор», «простой обмен», «простые вставки».

Лабораторная работа № 9. Алгоритмы поиска в линейном массиве.

Алгоритм простого перебора. Метод бинарного поиска.

Лабораторная работа № 10. Двумерные массивы.

Двумерные массивы. Способы описания массива. Заполнение массива случайным образом и с клавиатуры. Ввод и вывод элементов массива.

Лабораторная работа № 11. Решение задач на двумерные массивы.

Сумма матриц. Произведение матриц. Вывод на экран матрицы в принятом в математике виде.

Лабораторная работа № 12. Функции для работы с символьным типом.

Символьный тип. Функции для работы с символьным типом. Примеры и задачи.
Лабораторная работа № 13. Строковые величины. Функции для работы со строковыми.
Строковый тип. Операции над строковыми данными. Примеры и задачи.
Лабораторная работа № 14. Алгоритмы шифрования данных.
Строковый тип. Операции над строковыми данными. Примеры и задачи.
Лабораторная работа № 15. Подпрограмма-процедура.
Понятие подпрограммы. Отличие процедуры от функции. Вызов процедуры и функции.
Лабораторная работа № 16. Подпрограмма-функция.
Понятие подпрограммы. Отличие процедуры от функции. Вызов процедуры и функции.
Лабораторная работа № 17. Параметры процедур и функций.
Формальные и фактические параметры. Передача параметра по ссылке и по значению. Вызов процедуры и функции.
Лабораторная работа № 18. Рекурсивные подпрограммы.
Понятие рекурсии. Примеры задач с использованием рекурсии. Ручное исполнение программы с рекурсией.
Лабораторная работа № 19. Структурированный тип данных запись.
Тип Запись. Операции над записями. Оператор присоединения. Примеры использования типа запись.
Лабораторная работа № 20-21. Массивы записей.
Массивы записей. Примеры программ с использованием массивов записей.
Лабораторная работа № 22. Тип множество.
Тип множество. Построение множества. Действия над множествами. Вывод множества на экран. Примеры решения задач с использованием типа множество.
Лабораторная работа № 23. Работа с файлами. Чтение и запись информации из файлов.
Файловые переменные. Способы открытия файла (чтение, запись, дозапись). Чтение и запись из файла.
Лабораторная работа № 24. Типизированные и текстовые файлы. Операции для работы с файлами.
Файлы с типом. Текстовые файлы. Организация чтения из текстового файла. Примеры программ.
Лабораторная работа № 25. Графические возможности среды программирования. Процедуры и функции модуля CRT.
Модуль CRT. Управление курсором. Управление цветом.
Лабораторная работа № 26. Процедуры и функции модуля Graph.
Процедуры и функции модуля Graph. Графические операторы. Графические примитивы.
Лабораторная работа № 27. Организация анимации средствами среды программирования.
Построение графиков функций. Анимация средствами языка.

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

В преподавании дисциплины «Программирование» используются разнообразные образовательные технологии как традиционные, так и с применением активных и интерактивных методов обучения.

Активные и интерактивные методы обучения:

- *Интерактивная лекция (тема №2, 4, 6, 7, 10, 11, 15, 24);*
- *Групповая дискуссия (тема № 11, 12);*
- *Ролевые игры (тема № 5, 8, 9);*
- *Мозговые штурмы (тема № 19, 20);*
- *Применение имитационных моделей (тема №11);*
- *Разбор конкретных ситуаций (тема №1-25).*

**6. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ,
ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И
УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ
СТУДЕНТОВ**

1 семестр

Текущий контроль успеваемости

Рейтинг-контроль 1

1. Известно, что переменная, $a=8$. Соотнести оператор вывода с результатом, отображающимся на экране

writeln('a');	a=5
writeln(a);	5
writeln('a=',a);	a

2. Определите значения переменных a, b, c после выполнения фрагмента алгоритма

```
a:= 5;
b:=16;
a:=2 *b-4;
a:=a mod 6;
c:=sqrt(a)+3;
a:=a+b div 3;
```

3. Запишите формулу в линейном виде по правилам Pascal

$$z = \frac{2x + y^2}{x + 1} - \sqrt{x^2 + 2}$$

4. Восстановите формулу по её линейной записи

$$u := (2 * a + 4) / (\text{sqr}(a + 1) - 4) + 3 / \text{sqrt}(b).$$

5. Выберите фрагменты, где представлена команда ветвления в полной форме

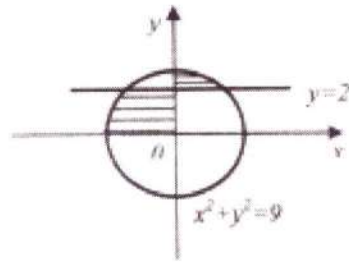
- a) if $x > y$ then $x := 0$ else $y := 0$;
- b) if $x > y$ then
begin
x := 0; y := 0;
end;
- c) if $x > y$ then $x := 0$;
- d) if $x > y$ then
begin
x := 0; y := 0;
end
else $y := 110$;

6. Определите значение переменных a и b в конце выполнения фрагмента алгоритма. Запишите, какая информация будет отображена на экране после выполнения фрагмента.

```
var a, b: integer;
begin
a := 7;
b := 12;
a:=b mod a;
if a>=b then
begin
a:=a-b; b:=b+2*a;
end
else
a:=sqr(a);
b:=bdiv a;
writeln(b, ' ', a);
end.
```

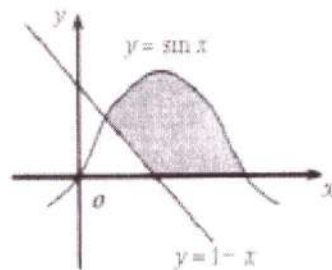
7. Определить логические значения сложных условий

- a) $(5 < 7)$ and not $(8 > 11)$ c) $(2 > 6)$ or $(7 < 6)$
 b) not $((6 > 5)$ or $(3 > 1))$ d) $(2 < 4)$ and $(5 > 11)$
 8. Записать условие попадания точки с координатами (x, y) в закрашенную область



9. Требовалось написать программу, которая вводит с клавиатуры координаты точки на плоскости $(x, y - \text{действительные числа})$ и определяет принадлежность точки заштрихованной области. Программист торопился и написал программу неправильно. Укажите пару чисел, при которых программа будет работать неправильно. Напишите фрагмент верного кода.

```
readln{x, y};
if y <= sin(x) then
  if y >= 1 - x then
    if y >= 0 then
      write('принадлежит')
    else
      write('не принадлежит')
  end
end.
```



Укажите пару чисел (x, y) , при которых программа будет работать неправильно. Напишите, как исправить работу программы (напишите фрагмент верного кода).

Рейтинг-контроль 2

1. Выполните вручную алгоритм и определите, что будет напечатано в результате работы следующего фрагмента программы:

```
Var n, s: integer;
begin
  n := 1;        s := 0;
  while n <= 101 do begin
    s := s + 7; n := n + 1
  end;
  write(s)
end.
```

2. Записать похожие характеристики в циклах с предусловием и постусловием.
 3. Ниже записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа: a и b . Укажите наименьшее из таких чисел x , при вводе которых алгоритм печатает сначала 2, а потом 15.

```
var x, a, b: integer;
begin
  readln(x);
  a := 0; b := 1;
  while x > 0 do
  begin
    a := a + 1;
    b := b * (x mod 10);
    x := x div 10
  end;
  writeln(a); write(b)
end.
```

4. Изменить приведенный алгоритм, записав вместо цикла while цикл repeat так, чтобы

результат работы алгоритма не изменился.

```
s:=0; readln(a);  
while a>0 do  
begin  
if a mod 2 =0 then s:=s+a;  
readln(a);  
end;
```

5. Записать фрагмент алгоритма, который вводит в клавиатуры числа, а затем каждое число умножает на 2 и выводит на экран. Ввод последовательности оканчивается нулем.

Рейтинг-контроль 3

1. Выполните вручную фрагмент алгоритма. Вычислите значение переменной s.

s: = 5 ;					
for i:=3 to 6 do					
s: = s + i*2;					

Укажите: параметр цикла, начальное значение, конечное значение, количество повторов цикла, значение переменной s.

2. Перепишите фрагмент алгоритма, заменив цикл for циклом while так, чтобы результат работы алгоритма не изменился

```
к: = 10;  
for j:=10 to 100 do  
begin  
x:=j*j;  
к:=к+2*x;  
end;  
write(k);
```

3. Выполните фрагмент алгоритма. В ответе укажите, что будет выведено на экране после его выполнения.

```
a: = 150;  
for i:=8 downto 6 do  
begin  
a:=a-sqr(i);  
write(a);  
end;
```

4. Описать массив b, состоящий из 15 элементов. Элементы массива являются вещественными числами.

5. Написать фрагмент кода, задающий для элементов описанного в задании 1 массива b случайные значения в диапазоне [-20; 20].

6. Написать фрагмент программы для задания элементов массива по следующему правилу: каждый элемент массива равен квадрату своего индекса, увеличенному на 1.

7. В таблице Dat хранятся оценки (по десятибалльной шкале) студента по программированию за 8 прошедших с начала учёбы семестров. Определите, что будет напечатано в результате выполнения следующего алгоритма.

```
Var k, ш, term: integer;  
Dat: array[1...8] of integer;  
Begin  
Dat[1]:=7;            Dat[2]:=7;Dat[3]:=6;    Dat[4]:=8;  
Dat[5]:=4;            Dat[6]:=5;Dat[7]:=8;    Dat[8]:=7;  
term:=1; m := Dat[1];  
for к :=2 to 8 do  
if Dat[k] > m then  
begin  
m: = Dat[k];  
term := k; end;  
write(term);
```

End.

8. В программе описан одномерный целочисленный массив с индексами от 0 до 10. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется. Чему будут равны элементы данного массива в конце выполнения алгоритма.

```
for i:=0 to 10 do
  A[i]:=i;
for i:=0 to 4 do
begin
  k:=A[2 * i];
  A[2*i]:=A[2*i+1];
  A[2*i+1]:= k;
end;
```

Промежуточная аттестация по итогам освоения дисциплины (зачет).

1. Алгоритм, свойства алгоритма.
2. Способы записи алгоритма.
3. Исполнители алгоритмов.
4. Языки программирования и среды программирования.
5. Способы отладки программ.
6. Среда программирования Pascal. Структура программы. Описание констант и переменных.
7. Объявление переменных в программе. Схема типов данных в Pascal.
8. Числовые целые и вещественные типы данных. Логический тип.
9. Линейные алгоритмы. Операторы ввода и вывода в Pascal. Оператор присваивания.
10. Команда ветвления в полной и неполной форме (применение, блок-схема, синтаксис в Pascal).
11. Цикл с параметром (применение, блок-схема, синтаксис в Pascal).
12. Цикл с предусловием (применение, блок-схема, синтаксис в Pascal).
13. Цикл с постусловием (применение, блок-схема, синтаксис в Pascal).
14. Вложенные циклы.

Примерные задания для самостоятельной работы студентов

1. Составьте блок-схемы для решения задач лабораторных работ.
2. Опишите компоненты структуры программы в Pascal (их назначение, синтаксис, примеры использования).
3. Составьте задания, для решения которых необходимо использовать команду ветвления в полной или неполной форме. Напишите программу для решения составленной задачи.
4. Составьте задания, для решения которых необходимо использовать циклический алгоритм. Напишите программу для решения составленной задачи с использованием различных видов циклов.
5. Составьте задания, для решения которых необходимо использовать циклический алгоритм с вложенным ветвлением. Напишите программу для решения составленной задачи.

2 семестр

Текущий контроль успеваемости

Рейтинг-контроль 1

1. Напишите фрагмент программы, который вводит с клавиатуры матрицу размером 3 x 4, а затем выводит её на экран в табличной форме.
2. Напишите программу, которая задает квадратную матрицу размера n так, что каждый элемент равен удвоенной сумме своих индексов.
3. Дан фрагмент программы, обрабатывающей двухмерный массив A[1..5, 1..4]:

```
k=4 ;
for m:=1 to 4 do
begin
  k:=k+1;
  for n:=1 to 5 do
  begin
    k:=m-k;
```

```

A[n, m] := n*n+m*m-2*k;
end;
end;

```

Чему будет равно значение A[3,1]?

4. Элементы двумерного массива A размером 10x10 первоначально были равны 1. Затем значения некоторых из них меняют с помощью следующего фрагмента программы:

```

for n:=1 to 4 do
for k:=1 to n+1 do
begin
A[n, k] := A[n, k]-1;
A[n, k+1] := A[n, k]-1;
end;

```

Запишите получившийся двумерный массив и укажите, сколько элементов массива в результате будут равны 0?

Рейтинг-контроль 2

1. Выполнить последовательность операторов для строки a. Определить значение переменных a, n, c, p после выполнения этих операторов.

```

c:=copy(a,3,5);
delete(a,4,2);
n:=Length(a);
p:=pos('o',a);
a[n-p]:=c[2];

```

2. Опишите функции для работы со строковыми величинами (назначение функции, имя функции, необходимые параметры, способы вызова функции, пример): delete, insert, inttostr, strtoint;
3. Напишите программу, которая меняет местами трети слова из 12 букв следующим образом: первую треть размещает на месте третьей, вторую - на месте первой, третью - на месте второй. Дана строка, состоящая из слов, разделенных пробелами. Вывести на экран в столбик те слова, которые имеют четную длину.

Рейтинг-контроль 3

1. Определение функции. Отличие функции от процедуры.
2. Передача параметров по значению.
3. Перечислите номера верных высказываний:
- Порядок следования фактических параметров подпрограммы не имеет значения
 - Глобальные параметры доступны в любом месте программы, включая подпрограммы
 - При передаче параметров по значению их значение в основной программе изменится после возвращения из подпрограммы
 - В подпрограмме-функции должен быть оператор, присваивающий некоторое значение имени функции
 - При вызове функции необходимо имя функции (с фактическими параметрами) указывать внутри некоторого оператора.
 - Фактические параметры процедуры и их типы перечисляются при описании процедуры, а формальные параметры - при вызове.
4. Определить значения переменной p в конце выполнения программы. Указать, какие значения будут выведены в конце выполнения программы (порядок вывода сохранить).

```

program example;
var m,n,p:integer;
procedure task(var a:integer; b:integer);
var irbyte; s: integer;
begin

```

```

for i:=a to b do
s:=s + i;
a:=s;
b:=a+b;
writeln(a, ", b);
end;
begin
m:= 3;
n:= 7;
task(m, n);
writeln(m, ' ', n);
p:=m+n;
end.

```

5. Дан массив слов. Для каждого слова необходимо вычислить и вывести количество слогов. Для подсчета количества слогов написать соответствующую подпрограмму.
6. С клавиатуры вводится матрица размера $n \times 3$. Для тройки элементов из каждой строки проверить. Образуют ли они пифагорову тройку, написав для этого соответствующую подпрограмму (числа a , b , c образуют пифагорову тройку, если $a^2 + b^2 = c^2$).

Промежуточная аттестация по итогам освоения дисциплины (экзамен).

1. Массивы в Pascal (определение, описание массивов, ввод и вывод массивов).
2. Массивы в Pascal. Сортировка методом простого выбора.
3. Массивы в Pascal. Сортировка пузырьковым методом.
4. Массивы в Pascal. Алгоритмы линейного и бинарного поиска элемента в массиве.
5. Двумерные массивы - матрицы (назначение, описание матриц, ввод и вывод матриц, примеры использования).
6. Символьный тип. Функции для работы с символами.
7. Строковые величины. Функции для работы со строками.
8. Подпрограммы. Функции.
9. Подпрограммы. Процедуры.
10. Подпрограммы. Локальные и глобальные параметры. Передача параметров структурированных типов.
11. Подпрограммы. Передача параметров по имени и по значению.
12. Рекурсивные подпрограммы.

Примерные задания для самостоятельной работы студентов

1. Приведите пример необходимости использования структурированного типа данных - массива. Составьте и решите задачу, требующую организации данных в виде массива.
2. Приведите пример необходимости использования структурированного типа данных - двумерного массива. Составьте и решите задачу, требующую организации данных в виде двумерного массива.
3. Изучите дополнительные виды сортировок (пирамидальная, быстрая и т.д.). Напишите программы, реализующие данные алгоритмы сортировок.
4. Создайте программу для осуществления шифровки и дешифровки сообщения выбранным вами способом.
5. Реализуйте программы ранее решенных задач с использованием подпрограмм.
6. Рассмотрите примеры задач, для решения которых необходима реализация рекурсивных подпрограмм (игра Баше, Ханойская башня и т.д.). Составьте программы для решения этих задач.

3 семестр

Текущий контроль успеваемости

Рейтинг-контроль I

1. Описать тип `avto`, содержащий информацию о марке автомобиля, годе выпуска, пробеге и

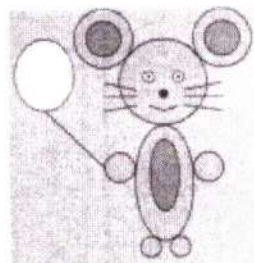
- регистрационном номере. Описать две переменных данного типа.
2. Описать массив из 10 элементов, состоящий из записей, представленных в задании 1.
 3. Написать фрагмент программы, который осуществляет ввод массива из задания 2 с клавиатуры.
 4. Написать оператор, который для 8-го элемента описанного в задании 2 массива задает год выпуска 2008.

Рейтинг-контроль 2

1. Напишите фрагмент кода, который создает на диске C новый файл с именем digite.txt и заполняет его числами от 0 до 9.
2. Опишите назначение и синтаксис функции assign.
3. Опишите назначение и синтаксис функции reset.
4. Опишите назначение и синтаксис функции append.
5. Опишите назначение и синтаксис функции rewrite.
6. Дан типизированный файл целых чисел. Оставить в файле наибольшее из чисел, удалив остальные.

Рейтинг-контроль 3

1. Охарактеризуйте инструмент Pen (назначение, характеристики). Приведите примеры изменения характеристик данного инструмента.
2. Опишите методы для рисования следующих графических примитивов (с указанием параметров):
 - a. прямоугольник
 - b. эллипс
 - c. вывод текста в графическом режиме.
3. Составить программу, рисует три треугольника: равносторонний, равнобедренный и разносторонний (координаты треугольников выберите самостоятельно). Цвет контура и заливки у каждого треугольника должен быть отличным от остальных. Под каждым рисунком должна быть подпись с указанием типа треугольника.
4. Создать изображение средствами графики в Pascal



Промежуточная аттестация по итогам освоения дисциплины (экзамен).

1. Тип запись. Назначение, описание, ввод-вывод, обращение к записям.
2. Оператор with.
3. Массивы записей.
4. Файлы. Описание файлов, операции для работы с файлами.
5. Файлы. Запись информации в файл, чтение информации из файла.
6. Текстовые файлы. Описание, операции для работы с текстовыми файлами.
7. Типизированные файлы. Функции для работы с типизированными файлами.
8. Графические возможности модуля crt.
9. Модуль GraphABC. Окно модуля. Основные принципы работы.
10. Графические примитивы модуля GraphABC (точка, прямая, эллипс, прямоугольник, дуга).
11. Вывод текста в окне Graph. Параметры текста.
12. Объект Pen. Назначение. Параметры объекта Pen.
13. Объект Brush. Назначение. Параметры объекта Brush.
14. Построение графиков функции средствами модуля GraphABC.
15. Анимация в среде программирования PascalABC.

Примерные задания для самостоятельной работы студентов

1. Составьте задачи, для решения которых необходимо ввести структурированный тип запись. Напишите программу для решения задачи.
2. Для составленных ранее задач организуйте ввод и вывод информации с использованием файлов.

3. Создайте и выполните с помощью языка программирования изображение, включающее в себя как можно большее количество графических примитивов.
4. Составьте программу, анимирующую один из физических процессов (на выбор студента).

Фонд оценочных средств для проведения аттестации уровня сформированности компетенций обучающихся по дисциплине оформляется отдельным документом.

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

7.1. Книгообеспеченность

Наименование литературы: автор, название, вид издания, издательство	Год издания	КНИГООБЕСПЕЧЕННОСТЬ	
		Количество экземпляров изданий в библиотеке ВлГУ в соответствии с ФГОС ВО	Наличие в электронной библиотеке ВлГУ
1	2	3	4
Основная литература			
1. Тишин, В. И. Программирование на Паскале [Электронный ресурс]: практикум / В. И. Тишин. - Эл. изд. - М.: БИНОМ. Лаборатория знаний. - 364 с.: ил.	2013		http://7.nanium.com/bookread2.php?book=500900
2. Грацианова, Т.Ю. Программирование в примерах и задачах [Электронный ресурс / Т.Ю). Грацианова. - М.: БИНОМ. - (ВМК МГУ - школе)	2015		http://www.studentlibrary.ru/book/ISBN9785996327812.html
3. Комлев, Н.Ю. Самоучитель игры на Паскале. ABC и немного Турбо. [Электронный ресурс] / Комлев Н.Ю. - М.: СОЛОН- ПРЕСС. -256 с.: ил.	2013		http://www.studentlibrary.ru/book/ISBN9785913591128.html
Дополнительная литература			
1. Окулов, С.М. Программирование в алгоритмах [Электронный ресурс] / С.М. Окулов. - М.: БИНОМ. - Электронное издание на основе: Программирование в алгоритмах [Электронный ресурс] / С.М. Окулов. - 5-е изд. (эл.). - М.: БИНОМ. Лаборатория знаний, -383 с.: ил.	2014		http://www.studentlibrary.ru/book/ISBN9785996323111.html
2. Канцедал, С.А. Алгоритмизация и программирование: Учебное пособие / С.А. Канцедал. - М.: ИД ФОРУМ: НИЦ ИНФРА-М. - 352 с.: ил.; - (Профессиональное образование).	2014		http://znanium.com/cataloga.php?bookinfo=429576
Давыдова Н.А. Программирование [Электронный ресурс]: учебное пособие/ Давыдова Н.А., Боровская Е.В. - Электрон, текстовые данные. - М.: БИНОМ. Лаборатория знаний, -239 с.	2014		http://www.iprbookshop.ru/6485

7.2. Периодические издания

Журнал «Информатика и образование»: <http://infojournal.ru/>

7.3. Интернет-ресурсы

1. msdn.microsoft.com.
2. <http://projecteuler.net/>.
3. <http://pascalabc.net/>

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для реализации данной дисциплины имеются специальные помещения для проведения занятий лекционного типа, лабораторных работ, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы.

Вид учебных занятий по дисциплине	Наименование специальных помещений и помещений для самостоятельной работы	Оснащенность специальных помещений и помещений для самостоятельной работы	Перечень лицензионного программного обеспечения. Реквизиты подтверждающего документа
1	2	3	4
Лекционные занятия	Лаборатории информатики и информационных технологий в образовании, компьютерные классы. Владимир, пр-т. Строителей, д. 11, (к.7) Аудит. 226, 241, 242, 243. Лекционно-семинарская аудитория. Владимир, пр-т. Строителей, д. 11, (к.7)	Компьютерный класс на основе ЭВМ ПК IntelCore с доступом в сеть Интернет, маркерная и интерактивная доски, переносной ноутбук, наушники, колонки. Мультимедийный комплекс в составе: Ноутбук с выходом в сеть Интернет, мультимедиа проектор, экран белый матовый, доска маркерная, доска меловая.	Лицензии на Microsoft Windows/Office: Microsoft Open License 61248656/62857078/6384836 8/64196124 Visual Studio professional: MSDN подписка. Mathcad 14.0 M011: PKG-7518-FN Лицензия наантивирусное ПО: Kaspersky Endpoint Security Standart 1356-161220-101943-827-71
Лабораторные работы	Лабораторные занятия проводятся в компьютерных классах с возможностью индивидуальной работы каждого студента за компьютером.		

Рабочую программу составил
доцент кафедры МОиИТ _____

С. Кау

Наумова С.Б.

Рецензент _____

*зам. директора по УВР
Л150 У. Ермакова И. В.*

Ермаков

Ермакова И. В.

Программа рассмотрена и одобрена на заседании кафедры МОиИТ.

Протокол № 10 от 29.06.2018 года

Заведующий кафедрой МОиИТ _____

Ю. Е.

Евсеева Ю.Ю.

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии
направления 44.03.05 Педагогическое образование.

Протокол № 1 от 28.08.2018 года.

Председатель комиссии _____

директор ПМ Артамонова М. В.

Артамонова

**ЛИСТ ПЕРЕУТВЕРЖДЕНИЯ
РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ**

Рабочая программа одобрена на _____ учебный год

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на _____ учебный год

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____

Рабочая программа одобрена на _____ учебный год

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой _____