

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



УТВЕРЖДАЮ  
Проректор по ОД

А.А.Панфилов

« 20 » 01 \_\_\_\_\_ 2017г.

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ  
ОБЩЕПРОФЕССИОНАЛЬНОГО ЦИКЛА  
ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ**  
для специальности среднего профессионального образования  
**технического профиля**  
**15.02.10 Мехатроника и мобильная робототехника (по отраслям)**

Владимир, 2017

Программа учебной дисциплины разработана на основе Федерального государственного образовательного стандарта (далее - ФГОС) по специальности среднего профессионального образования (далее СПО) 15.02.10. Мехатроника и мобильная робототехника (по отраслям), утвержденного приказом Министерства образования и науки РФ от 09.12.2016 г. № 1550) по специальности

Кафедра-разработчик: МиЭСА

Рабочую программу составил: доцент кафедры МиЭСА



Е.В. Еропова

Программа рассмотрена и одобрена на заседании кафедры МиЭСА

протокол № 6 от «09» 01 2017 года

Заведующий кафедрой МиЭСА  А.А.Кобзев

Программа рассмотрена и одобрена на заседании УМК КИТП

протокол № 7 от «20» января 2017 года

Директор КИТП  Ю.Д. Корогодов

	стр.
<b>1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	4
<b>2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	5
<b>3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	13
<b>4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ</b>	14

# **1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ**

## **2. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ**

### **1.1. Область применения рабочей программы**

Рабочая программа учебной дисциплины является частью основной профессиональной образовательной программы в соответствии с ФГОС по специальности 15.02.10. Мехатроника и мобильная робототехника (по отраслям)

Рабочая программа учебной дисциплины может быть использована в дополнительном профессиональном образовании и в профессиональной подготовке работников в области мехатроники и робототехники при наличии полного (среднего) образования.

### **1.2. Место учебной дисциплины в структуре основной профессиональной образовательной программы:**

Дисциплина «Основы алгоритмизации и программирования» относится к блоку дисциплин естественнонаучного цикла в соответствии с ФГОС СПО, При изучении дисциплины используются знания, полученные в курсе «Математика», «Информационные технологии». Содержание дисциплины является основой для освоения дисциплин «Вычислительная техника», «Основы теории автоматического управления».

В учебном плане предусмотрены теоретические занятия – лекции, лабораторные работы, самостоятельная работа студентов.

### **1.3. Цели и задачи учебной дисциплины – требования к результатам освоения учебной дисциплины:**

Целями освоения дисциплины «Основы алгоритмизации и программирования» получение информации о логике построения программы, принципах составления программ; получение представления о том, как делаются профессиональные программные продукты понимание основных положений и принципов составления алгоритма и разработки программ, основных принципов, метод и свойств информационных и телекоммуникационных технологий в профессиональной деятельности, получение навыка обработки и анализа информации с применением программных средств и вычислительной техники, навыка создания завершённой прикладной программы.

В результате освоения программы подготовки специалистов среднего звена техник должен обладать общими компетенциями, включающими в себя способность:

ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие.

ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей.

ОК 09. Использовать информационные технологии в профессиональной деятельности.

Техник должен обладать профессиональными компетенциями, соответствующими видам деятельности:

ПК 1.1. Выполнять монтаж компонентов и модулей мехатронных систем в соответствии с технической документацией.

ПК 1.3. Разрабатывать управляющие программы мехатронных систем в соответствии с техническим заданием.

В результате освоения дисциплины обучающийся должен уметь:

- использовать языки программирования;
- строить логически правильные и эффективные программы;
- составлять простые блок-схемы алгоритмов;
- составлять программы на алгоритмическом языке высокого уровня;
- работать в интегрированной среде изучаемых языков программирования;
- производить тестирование программного продукта на выявление ошибок.

знать:

- общие принципы построения алгоритмов, основные алгоритмические конструкции;
- понятие системы программирования;
- подпрограммы;
- основные приемы программирования;
- интегрированные среды изучаемых языков программирования.

владеть:

- основными элементами процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти;
- основными приемами составления программ.

#### 1.4. Количество часов на освоение программы учебной дисциплины:

максимальной учебной нагрузки обучающегося **96** часов, в том числе:  
обязательной аудиторной учебной нагрузки обучающегося **64** часов;  
самостоятельной работы обучающегося **32** часов.

## 2. СТРУКТУРА И ПРИМЕРНОЕ СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

### 2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Максимальная учебная нагрузка (всего)	<b>96</b>
Обязательная аудиторная учебная нагрузка (всего)	<b>64</b>
в том числе:	
лекции	32
лабораторные работы	32
практические занятия	
контрольные работы	
Самостоятельная работа обучающегося (всего)	<b>32</b>
в том числе:	
работа с информационными источниками	10
подготовка отчетов по лабораторным работам	10
<b>Итоговая аттестация в форме</b>	<b>дифференцированный зачет</b>

**а. Примерный тематический план и содержание учебной дисциплины  
ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ**

Статья I. 2.2. Примерный тематический план и содержание учебной дисциплины <u>Основы алгоритмизации и программирования</u>			
<i>наименование</i>			
Наименование разделов и тем	Содержание учебного материала, лабораторные и практические работы, самостоятельная работа обучающихся, курсовая работ (проект) (если предусмотрены)	Объем часов	Уровень освоения
1	2	3	4
<b>Раздел 1.</b>	<b>Основные принципы алгоритмизации и программирования</b>		
<b>Тема 1.1</b> Основные понятия алгоритмизации	<b>Содержание учебного материала:</b> Понятие алгоритма. Свойства алгоритмов. Формы записей алгоритмов. Общие принципы построения алгоритмов. Основные алгоритмические конструкции: линейные, разветвляющиеся, циклические. Данные: понятие и типы. Основные базовые типы данных и их характеристика. Структурированные типы данных и их характеристика. Методы сортировки данных.	<b>6</b>	<b>1</b>
	1 Введение. Предмет и задачи дисциплины основы алгоритмизации и программирования		
	2 Понятие программы и программного обеспечения		
	3 Понятие алгоритма. Свойства алгоритмов.		
	4 Понятие и элементы блок-схем.		
	5 Проектирование блок-схем.		
	6 Линейный алгоритм.		
	7 Разветвляющийся алгоритм. Полная и не полная конструкция.		
	8 Циклический алгоритм. Полная и не полная конструкция. Цикл с пред условием.		
	9 Циклический алгоритм. Цикл с пост условием.		
	<b>Самостоятельная работа обучающихся:</b> 1. Изучение тем:  1)История развития предмета основы алгоритмизации и программирования. 2)Происхождения понятия «Алгоритм». 3)Классификация программного обеспечения. Результат: сообщение на уроке по предложенным темам.	<b>4</b>	<b>2</b>

	2. Решить систему уравнений $ax+by+c=0$ и $a_1x+b_1y+c_1=0$ графически. Результат: вывод результатов на бумажном носителе. 3. Изучение материала лекции, чтение и анализирование текста.		
<b>Тема 1.2</b> Языки и системы программирования	<b>Содержание учебного материала:</b> Эволюция языков программирования. Классификация языков программирования. Элементы языков программирования. Понятие системы программирования. Исходный, объектный и загрузочный модули. Интегрированная среда программирования.	<b>6</b>	<b>1</b>
	11 Основные понятия языка программирования. Среда программирования.		
	12 Интерфейс среды программирования.		
	13 Данные. Типы данных.		
	14 Арифметические выражения. Стандартные функции.		
	15 Организация программ линейной структуры. Структура оператора на языке Паскаль.		
	16 Организация программы разветвляющей структуры. Условный оператор If Then Else		
	<b>Лабораторные работы</b>	<b>6</b>	<b>3</b>
	1 Знакомство со средой программирования Turbo Pascal.		
	2 Ввод и отладка простейших линейных программ. Изучение работы среды программирования Turbo Pascal.		
	3 Ввод и отладка простейших линейных программ.		
	4 Целочисленная арифметика. Задачи на целочисленное деление.		
	5 Задачи на построение и расчет математических выражений.		
	<b>Самостоятельная работа обучающихся:</b> 1. Примерная тема доклада: 1) История языков программирования. 2) Выдающиеся ученые в области программирования. 3) Современные ученые и языки программирования. Результат: сообщение на уроке по предложенным темам. 2. Решение задач и представление результата на бумажном носителе. 1. Составить программу для нахождения суммы двух чисел. 2. Составить программу для нахождения суммы четырех чисел. 3. Найти значения выражения: $(a+(d-12)*3)*(c-5*k)$ , где значение вводятся с клавиатуры. 4. Написать программу вывода на экран нескольких чисел в виде 13 14 15 16	<b>6</b>	<b>3</b>

	3. Повторение лекции перед выполнением практической работы.																																											
<p style="text-align: center;"><b>Тема 1.3</b> Логические основы алгоритмизации</p>	<p><b>Содержание учебного материала:</b> Основы алгебры логики. Логические операции с высказываниями: конъюнкция, дизъюнкция, инверсия. Законы логических операций. Таблицы истинности.</p>	<b>4</b>	<b>1</b>																																									
	17   Логические операции и выражения																																											
	<b>Лабораторные работы</b>	<b>2</b>	<b>3</b>																																									
	6   Логические операции и выражения.																																											
	<p><b>Самостоятельная работа обучающихся:</b> 1. Примерная тема доклада:  Статья П. Логические элементы и их история Результат: сообщение на уроке по предложенным темам. 2. Решение задач <b>1.</b> В математической логике известна функция Вебба, или стрелка Пирса, <math>(x \downarrow y)</math> ее таблица истинности имеет вид:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>X</td><td></td><td><math>X \downarrow Y</math></td></tr> <tr><td>False</td><td>False</td><td>True</td></tr> <tr><td>False</td><td>True</td><td>False</td></tr> <tr><td>True</td><td>False</td><td>False</td></tr> <tr><td>True</td><td>True</td><td>False</td></tr> </table> <p>Проверьте, что <math>x \downarrow y</math> эквивалентно <math>\text{Not}(x) \text{ Or } \text{Not}(y)</math>. Составьте программу проверки эквивалентности этих двух логических функций. 2. Дана логическая функция, например, <math>(x \rightarrow y) \rightarrow z</math>. Построить таблицу истинности данной функции. Схема построения приведена в таблице. В первом столбике приведены возможные значения наборов переменных <math>x, y</math> и <math>z</math> (значение True обозначено как единица, значение False – как нуль).</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr><th>XYZ</th><th><math>X \rightarrow Y</math></th><th><math>(X \rightarrow Y) \rightarrow Z</math></th></tr> </thead> <tbody> <tr><td>000</td><td>1</td><td>0</td></tr> <tr><td>001</td><td>1</td><td>1</td></tr> <tr><td>01</td><td>1</td><td>0</td></tr> <tr><td>011</td><td>1</td><td>1</td></tr> <tr><td>100</td><td>0</td><td>1</td></tr> <tr><td>101</td><td>0</td><td>1</td></tr> <tr><td>110</td><td>1</td><td>0</td></tr> <tr><td>111</td><td>1</td><td>1</td></tr> </tbody> </table> <p>Преобразуйте эту формулу в эквивалентную ей. Составьте программу проверки эквивалентности этих двух логических формул. Результат: вывод результатов на бумажном носителе.</p>	X		$X \downarrow Y$	False	False	True	False	True	False	True	False	False	True	True	False	XYZ	$X \rightarrow Y$	$(X \rightarrow Y) \rightarrow Z$	000	1	0	001	1	1	01	1	0	011	1	1	100	0	1	101	0	1	110	1	0	111	1	1	4
X		$X \downarrow Y$																																										
False	False	True																																										
False	True	False																																										
True	False	False																																										
True	True	False																																										
XYZ	$X \rightarrow Y$	$(X \rightarrow Y) \rightarrow Z$																																										
000	1	0																																										
001	1	1																																										
01	1	0																																										
011	1	1																																										
100	0	1																																										
101	0	1																																										
110	1	0																																										
111	1	1																																										



Раздел 2	Программирование на алгоритмическом языке Турбо-Паскаль			
Тема 2.1 Циклы	<b>Содержание учебного материала:</b> Оператор цикла FOR. Составление программ с использованием цикла FOR. Вложенные циклы. Логические циклы While и Repeat.		<b>8</b>	1,2
	18	Оператор цикла FOR		
	19	Составление программ с использованием цикла FOR To Do		
	20	Составление программ с использованием цикла FOR DownTo Do		
	21	Вложенные циклы		
	22	Решение задач с использованием вложенных циклов		
	23	Логический цикл While.		
	24	Решение задач с использованием цикла While.		
	25	Логический цикл Repeat.		
	26	Решение задач с использованием цикла Repeat.		
	<b>Лабораторные работы</b>		<b>10</b>	<b>2</b>
	7	Составление программ с использованием цикла FOR.		
	8	Составление программ с использованием цикла FOR		
	9	Вложенные циклы		
	10	Логический цикл While.		
11	Составление программ с использованием цикла While.			
12	Логический цикл Repeat .			
13	Составление программ с использованием цикла Repeat			
	<b>Самостоятельная работа обучающихся:</b> <b>1. Примерная тема доклада:</b> Цикл FOR. Цикл While. Цикл Repeat. Результат: сообщение на уроке по предложенным темам. <b>2.Выполнение домашней работы по составлению алгоритмов и программ.</b> <b>1. Дана последовательность операторов:</b> a:=1; b:=1; While a+b<8 Do Begin a:=a+1; b:=b+2 End; S:=a+b Сколько раз выполняется проверка логического выражения в операторе While? Определите значения переменных a, b, s после завершения этой последовательности операторов? <b>2. Определите значения переменных a и b после выполнения операторов:</b> a:=1; b:=1;		<b>6</b>	3

While a<=3 Do Begin a:=a+1; b:=b+1;

**3.** Определите значение переменной s после выполнения следующих операторов:

S:=0; i:=0 While i<5 Do Inc (i); s:=s+100 Div i;

S:=0; i:=1; While i>1 Do Begin s:=s+100 Div i;

Dec(i) End;

**4.** Дан фрагмент программы с ошибками (их не больше 5) вычисление факториала f числа n:

K:=1; f:=0;

While k<n Do f=f\*k

K:=k+1;

Найдите эти ошибки.

**5.** Найдите и исправьте ошибки в следующем фрагменте программы, определяющей для заданного натурального числа n число, записанное цифрами числа n в обратном порядке.

P:=n;

While p>=0 Do Begin

A:=a+p Mod 10;

P:=p Div 10

End;

Примечание

Задания 1-5 рекомендуется выполнять, используя режим ручной трассировки.

**6.** Найти минимальное число, больше 300, которое нацело делится на 19.

**7.** Приписать по 1 в начало и в конец записи числа n. Например, было n=3456, Стало n=134561.

**8.** Поменять местами первую и последнюю цифра числа n. Например, из числа 8547 должно быть получено число 7548.

**9.** Приписать к исходному числу n такое же число. Например, из числа 1903 должно быть получено число 19031903.

**10.** Определить, является ли заданное число степенью 3.

**11.** Составьте программу, проверяющую, является ли заданное натуральное число палиндромом, то есть таким, десятичная запись которого читается одинаково слева направо и справа налево.

Примечание

Задача отличается от ранее рассмотренной тем, что количество цифр в числе неизвестно, а из этого следует, что тип используемого цикла должен быть другой.

**12.** Выяснить, является ли последовательность цифр натурального числа при просмотре их справа налево возрастающей последовательностью. Например, для числа 76431 ответ положительный, для чисел 6331, 9782 – отрицательный.

**13.** Вводится последовательность целых ненулевых чисел, признак окончания ввода – ввод 0.

	<p>Количество чисел не меньше 2. Выяснить:</p> <ul style="list-style-type: none"> <li>• Является ли последовательность возрастающей;</li> <li>• Есть ли в ней хотя бы одна пара одинаковых (соседних) чисел;</li> <li>• Является ли последовательность знакопередающей (3, -2, 4, -5, 0 - Да; 5, -4, -7, 8, 0 – Нет).</li> </ul> <p><b>14.</b> Выяснить, сколько раз в натуральном числе встречается его максимальная цифра. Например, в числе 581088 – 3 раза, в числе 4537 – 1 раз.</p> <p><b>15.</b> Выяснить, является ли разность максимальной и минимальной цифр числа четной.</p> <p>Результат: вывод результатов на бумажном носителе.</p>		
<b>Тема 2.2</b> Массивы	<p><b>Содержание учебного материала:</b> Понятие массива. Одномерные и не одномерные массивы. Ввод и вывод массива. Нахождение элементов массива по заданным условиям. Сортировка элементов массива.</p>	4	3
	<p><b>Лабораторные работы</b></p>	8	2
	14   Работа с элементами массива.		
	15   Нахождение положительного и отрицательного элемента массива		
	16   Нахождение суммы и разности элементов массива		
	17   Формирование элементов массива путем ввода с клавиатуры		
	18   Нахождение наименьшего и наибольшего элементов массива		
	19   Сортировка массива методом Пузырька		
	20   Быстрая сортировка массива		
	21   Работа с двумерными массивами.		
<p><b>Самостоятельная работа обучающихся:</b> 1. Примерная тема доклада: 1) История происхождения понятия массив. 2) Принципы Джона Фон Неймана. Результат: сообщение на уроке по предложенным темам. 2. Выполнение домашней работы по составлению алгоритмов и программ. Результат: вывод результатов на бумажном носителе.</p>	4	3	
<b>Тема 2.3</b> Процедуры	<p><b>Содержание учебного материала:</b> Понятие и свойства процедур. Структура программы с помещением процедуры.</p>	2	1
	27   Понятие процедуры.		
	<p><b>Лабораторные работы</b></p>	4	2
	22   Помещение процедуры в программу		
	23   Составление и отладка программ с использованием процедур		
	24   Составление и отладка программ с использованием процедур		
<p><b>Самостоятельная работа обучающихся:</b> 1. Примерная тема доклада: Элементы теории вероятности и их связь с процедурами.</p>	4		

	Результат: сообщение на уроке по предложенным темам. 2. Составить программу вычисления $(\min(a,c)-\min(a,b))/(5+\min(b,c))$ Результат: вывод результатов на бумажном носителе.		
<b>Тема 2.4</b> Функции	<b>Содержание учебного материала:</b> Понятие и свойства функции. Описание функции. Составление программ с применением функций.	2	1
	<b>Лабораторные работы</b>	2	2
	25 Ввод функции		
	26 Составление и отладка программ с использованием функции		
	27 Составление и отладка программ с использованием функции		
	28 Составление и отладка программ с использованием функции		
	<b>Самостоятельная работа обучающихся:</b> 1. Подготовка отчётов по лабораторным работам	4	3
<b>Всего:</b>		<b>96</b>	

Для характеристики уровня освоения учебного материала используются следующие обозначения:

1. – ознакомительный (узнавание ранее изученных объектов, свойств);
2. – репродуктивный (выполнение деятельности по образцу, инструкции или под руководством)
3. – продуктивный (планирование и самостоятельное выполнение деятельности, решение проблемных задач)

### 3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ

#### 3.1. Материально-техническое обеспечение

Реализация программы дисциплины требует наличия учебного кабинета.

Оборудование учебного кабинета:

- посадочные места по количеству обучающихся;
- доска для письма;
- рабочее место преподавателя;
- презентационная техника (проектор, экран, компьютер/ноутбук);
- рабочие места студентов, оснащенные компьютерами с доступом в Интернет, предназначенные для работы в электронной образовательной среде.

#### 3.2. Информационное обеспечение обучения

**Перечень учебных изданий, Интернет-ресурсов, дополнительной литературы**

**Основные источники:**

1. Программирование на Паскале [Электронный ресурс] / Тишин В.И. - М. : БИНОМ, 2013. - <http://www.studentlibrary.ru/book/ISBN9785996321186.html>
2. Программирование в примерах и задачах [Электронный ресурс] / Т.Ю. Грацианова. - 2-е изд. (эл.). - М. : БИНОМ, 2015. - (ВМК МГУ - школе). - <http://www.studentlibrary.ru/book/ISBN9785996327812.html>
3. Алгоритмизация прикладных задач [Электронный ресурс] : учеб. пособие / А.И. Долгов - М. : ФЛИНТА, 2011. - <http://www.studentlibrary.ru/book/ISBN9785976500862.html>

**Дополнительные источники:**

1. "Delphi. Готовые алгоритмы [Электронный ресурс] / Стивенс Р. ; Пер. с англ. - М. : ДМК Пресс, 2007. - (Серия "Для программистов")." - <http://www.studentlibrary.ru/book/ISBN5940741061.html>
2. Учись писать, читать и понимать алгоритмы. Алгоритмы для правильного мышления. Основы алгоритмизации [Электронный ресурс] / Паронджанов В.Д. - М. : ДМК Пресс, 2012. - <http://www.studentlibrary.ru/book/ISBN9785940748007.html>
3. Самоучитель игры на Паскале. ABC и немного Турбо [Электронный ресурс] / Комлев Н.Ю. - М. : СОЛОН-ПРЕСС, 2015. - <http://www.studentlibrary.ru/book/ISBN9785913591630.html>

**Периодические издания:**

1. Научно-технический и научно-информационный журнал "Информационные технологии".

**Интернет-ресурсы:**

1. Сайт журнала «Вестник компьютерных и информационных технологий» - <http://www.vkit.ru/>.

#### 4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Контроль и оценка результатов освоения дисциплины осуществляется преподавателем в процессе проведения лабораторных и практических занятий, контрольных работ, а также выполнения обучающимися индивидуальных заданий.

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
<p>уметь:</p> <ul style="list-style-type: none"><li>– использовать языки программирования;</li><li>– строить логически правильные и эффективные программы;</li><li>– составлять простые блок-схемы алгоритмов;</li><li>– составлять программы на алгоритмическом языке высокого уровня;</li><li>– работать в интегрированной среде изучаемых языков программирования;</li><li>– производить тестирование программного продукта на выявление ошибок.</li></ul> <p>знать:</p> <ul style="list-style-type: none"><li>– общие принципы построения алгоритмов, основные алгоритмические конструкции;</li><li>– понятие системы программирования;</li><li>– основные элементы процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти;</li><li>– подпрограммы, составление библиотек программ;</li><li>– основные приемы программирования.</li><li>– интегрированные среды изучаемых языков программирования.</li></ul>	<p><i>Самостоятельные и контрольные работы, индивидуальные задания.</i></p> <p><i>Дифференцированный зачёт</i></p>