

**ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ
КАЛИНИНГРАДСКОЙ ОБЛАСТИ
ПРОФЕССИОНАЛЬНАЯ ОБРАЗОВАТЕЛЬНАЯ ОРГАНИЗАЦИЯ
«ХУДОЖЕСТВЕННО-ПРОМЫШЛЕННЫЙ ТЕХНИКУМ»**

УТВЕРЖДАЮ
Директор ГБУ КО ПОО «ХПТ»
Л. Н. Копцева
30.08.2017



**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
«ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»**

ОСНОВНАЯ ПРОФЕССИОНАЛЬНАЯ ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА
СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ –
ПРОГРАММА ПОДГОТОВКИ СПЕЦИАЛИСТОВ СРЕДНЕГО ЗВЕНА

09.02.05 Прикладная информатика (по отраслям)
(код наименования специальности)

базовой подготовки
(уровень подготовки)

основное общее образование
(уровень подготовки)

Форма обучения - очная

2017 г.

Согласовано
« » _____ 2017 г.
Заместитель директора по УМР
Павленко Г.Я. _____

Рассмотрено
на заседании отделения
общественных и гуманитарных дисциплин
Протокол № _____
« » _____ 2017 г.
Зав. отделением
Урывская Л.С. _____

Рабочая программа учебной дисциплины разработана на основе Федерального государственного образовательного стандарта среднего профессионального образования (ФГОС СПО) по специальности 230701 Прикладная информатика (по отраслям), МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ приказ от 21 июня 2010 г. N 643 .

Составитель: Нога А.И. – преподаватель ГБУ КО ПОО «ХПТ»

СОДЕРЖАНИЕ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	стр. 5
2. СТРУКТУРА И ПРИМЕРНОЕ СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	6
3. УСЛОВИЯ РЕАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	19
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	20

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

Основы алгоритмизации и программирования

1.1. Область применения программы

Рабочая программа учебной дисциплины является частью примерной основной профессиональной образовательной программы в соответствии с ФГОС по специальности СПО09.02.05 Прикладная информатика (по отраслям)

Рабочая программа учебной дисциплины может быть использована в программах дополнительного профессионального образования и профессиональной подготовке техников в области информационных технологий. Опыт работы не требуется.

1.2. Место дисциплины в структуре основной профессиональной образовательной программы:

Профессиональный цикл

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

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;
- взаимодействовать со специалистами смежного профиля при разработке, внедрению и применению объектов профессиональной деятельности;
- производить модификацию отдельных модулей программы;
- производить тестирование программного продукта на выявление ошибок.

В результате освоения дисциплины обучающийся должен **знать**:

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

- интегрированные среды изучаемых языков программирования.

1.4. Рекомендуемое количество часов на освоение программы дисциплины:

максимальной учебной нагрузки обучающегося **90** часов, в том числе:
 обязательной аудиторной учебной нагрузки обучающегося **60** часов;
 самостоятельной работы обучающегося **30** 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)	90
Обязательная аудиторная учебная нагрузка (всего)	60
в том числе:	
лабораторные занятия	
практические занятия	48
контрольные работы	4
курсовая работа (проект) <i>(если предусмотрено)</i>	
дифференцированный зачет	4
Самостоятельная работа обучающегося (всего)	30
в том числе:	
самостоятельная работа над курсовой работой (проектом) <i>(если предусмотрено)</i>	
Доклад..... Домашняя работа ...Оценка деятельности во время практических работ ...Индивидуальный опрос на уроке..... Указываются другие виды самостоятельной работы при их наличии (реферат, расчетно-графическая работа, домашняя работа и т.п.).	
Итоговая аттестация в форме дифференцированный зачет	

2.2. Примерный тематический план и содержание учебной дисциплины Основы алгоритмизации и программирования

<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
Раздел 1.	Основные принципы алгоритмизации и программирования	68	
Тема 1.1 Основные понятия алгоритмизации	Содержание учебного материала: 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.	2	1
	Самостоятельная работа обучающихся: 1. Примерная тема доклада: 1) История развития предмета основы алгоритмизации и программирования. 2) Происхождения понятия «Алгоритм». 3) Классификация программного обеспечения. Результат: сообщение на уроке по предложенным темам. 2. Решить систему уравнений $ax+by+c=0$ и $ax+by+c=0$ графически. Результат: вывод результатов на бумажные носители. 3. Изучение материала лекции, чтение и анализирование текста.	4	2
Тема 1.2 Языки и системы программирования	Содержание учебного материала: Эволюция языков программирования. Классификация языков программирования. Элементы языков программирования. Понятие системы программирования. Исходный, объектный и загрузочный модули. Интегрированная среда программирования.	2	1
	Практическое занятие	8	3
	1 Практическое занятие Знакомство со средой программирования TurboPascal.	2	3
	2 Практическое занятие Ввод и отладка простейших линейных программ. Изучение работы среды программирования TurboPascal. Ввод и отладка простейших линейных программ.	2	3
	3 Практическое занятие Целочисленная арифметика. Задачи на целочисленное деление.	2	3
	4 Практическое занятие Задачи на построение и расчет математических выражений.	2	3
Самостоятельная работа обучающихся: 1. Примерная тема доклада:	4	3	

	<p>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 5. Повторение лекции перед выполнением практической работы.</p>																									
<p style="text-align: center;">Тема 1.3 Логические основы алгоритмизации</p>	<p>Содержание учебного материала: Основы алгебры логики. Логические операции с высказываниями: конъюнкция, дизъюнкция, инверсия. Законы логических операций. Таблицы истинности.</p>	2	1																							
	<p>Практическое занятие 5 Практическое занятие Логические операции и выражения.</p>	1 1	3 3																							
	<p>Самостоятельная работа обучающихся: 1. Примерная тема доклада: Логические элементы и их история Результат: сообщение на уроке по предложенным темам. 2.Решение задач 1. В математической логике известна функция Вебба, или стрелка Пирса, $(x \downarrow y)$ ее таблица истинности имеет вид:</p> <table border="1" data-bbox="398 935 857 1118"> <tr><td>X</td><td></td><td>$X \downarrow Y$</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 $x \downarrow y$ эквивалентно $\text{Not}(x) \text{ Or } \text{Not}(y)$. Составьте программу проверки эквивалентности этих двух логических функций. 2. Дана логическая функция, например, $(x \rightarrow y) \rightarrow z$. Построить таблицу истинности данной функции. Схема построения приведена в таблице. В первом столбике приведены возможные значения наборов переменных x, y и z (значение True обозначено как единица, значение False – как нуль).</p> <table border="1" data-bbox="398 1342 987 1442"> <tr><td>XYZ</td><td>$X \rightarrow Y$</td><td>$(X \rightarrow Y) \rightarrow Z$</td></tr> <tr><td>000</td><td>1</td><td>0</td></tr> <tr><td>001</td><td>1</td><td>1</td></tr> </table>	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	2
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																								

	<table border="1"> <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> </table> <p>Преобразуйте эту формулу в эквивалентную ей. Составьте программу проверки эквивалентности этих двух логических формул. Результат: вывод результатов на бумажные носители.</p>	01	1	0	011	1	1	100	0	1	101	0	1	110	1	0	111	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	106																			
Тема 2.1 Циклы	Содержание учебного материала: Оператор цикла FOR. Составление программ с использованием цикла FOR. Вложенные циклы. Логические циклы While и Repeat.	2	1,3																		
	Практическое занятие	8																			
	6 Практическое занятие Составление программ с использованием цикла FOR.	2	3																		
	7 Практическое занятие Вложенные циклы. Логический цикл While.	2	3																		
	8 Практическое занятие Составление программ с использованием цикла While.	2	3																		
	9 Практическое занятие Логический цикл Repeat. Составление программ с использованием цикла Repeat	2	3																		
	<p>Самостоятельная работа обучающихся:</p> <p>1. Примерная тема доклада: Цикл FOR. Цикл While. Цикл Repeat. Результат: сообщение на уроке по предложенным темам.</p> <p>2. Выполнение домашней работы по составлению алгоритмов и программ.</p> <p>1. Дана последовательность операторов: 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?</p> <p>2. Определите значения переменных ai b после выполнения операторов: a:=1; b:=1; While a<=3 Do Begin a:=a+1; b:=b+1;</p> <p>3. Определите значение переменной спосле выполнения следующих операторов:</p>	8	3																		


```
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 цифру числа. Например, из числа 8547 должно быть получено число 7548.

9. Приписать к исходному числу 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. Количество чисел не меньше 2. Выяснить:

	<ul style="list-style-type: none"> • Является ли последовательность возрастающей; • Есть ли в ней хотя бы одна пара одинаковых (соседних) чисел; • Является ли последовательность знакопередающей (3, -2, 4, -5, 0 - Да; 5, -4, -7, 8, 0 – Нет). <p>14. Выяснить, сколько раз в натуральном числе встречается его максимальная цифра. Например, в числе 581088 – 3 раза, в числе 4537 – 1 раз.</p> <p>15. Выяснить, является ли разность максимальной и минимальной цифр числа четной.</p> <p>Результат: вывод результатов на бумажном носителе.</p>		
Тема 2.2 Массивы	Содержание учебного материала: Понятие массива. Одномерные и не одномерные массивы. Ввод и вывод массива. Нахождение элементов массива по заданным условиям. Сортировка элементов массива.	2	3
	Практическое занятие	12	3
	10 Практическое занятие Работа с элементами массива.	2	3
	11 Практическое занятие Нахождение положительного и отрицательного элемента массива	2	3
	12 Практическое занятие Нахождение суммы и разности элементов массива	2	3
	13 Практическое занятие Формирование элементов массива путем ввода с клавиатуры	2	3
	14 Практическое занятие Нахождение наименьшего и наибольшего элементов массива	2	3
	15 Практическое занятие Быстрая сортировка массива. Работа с двумерными массивами.	2	3
Самостоятельная работа обучающихся: 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.	4	3	
Тема 2.3 Процедуры	Содержание учебного материала: Понятие процедуры Помещение процедуры в программу	2	3
	Практическое занятие	4	3
	16 Практическое занятие Помещение процедуры в программу	2	3
	17 Практическое занятие Составление и отладка программ с использованием процедур	2	3
	Самостоятельная работа обучающихся: 1. Примерная тема доклада: Элементы теории вероятности и их связь с процедурами. 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.	2	3
Тема 2.4	Содержание учебного материала:		3

Функции	Практическое занятие		8	3
	18	Практическое занятие Понятие и свойства функции. Описание функции. Ввод функции в компьютер	2	3
	19	Практическое занятие Составление и отладка программ с использованием функции	2	3
	20	Практическое занятие Составление и отладка программ с использованием функции	2	3
	21	Практическое занятие Составление и отладка программ с использованием функции	2	3
	Самостоятельная работа обучающихся: 1. Примерная тема доклада: Создание / Уничтожение объектов в языках программирования. Результат: сообщение на уроке по предложенным темам. 2. Определить является ли данное слово перевертышем. Результат: вывод результатов на бумажном носителе.		2	3
Тема 2.5 Рекурсия	Содержание учебного материала:			3
	Практическое занятие		2	3
	22	Практическое занятие Понятие рекурсии Составление задач рекурсивного типа	2	3
	Самостоятельная работа обучающихся: 1. Примерная тема доклада: Динамические структуры объектов Результат: сообщение на уроке по предложенным темам. 2. Решить систему уравнений $\{ax+by+c=0 \text{ и } a_1x+b_1y+c_1=0$. Результат: вывод результатов на бумажном носителе.		2	3
	Содержание учебного материала:			3
Тема 2.6 Графика	Практическое занятие		4	3
	23	Практическое занятие Построение графических объектов в языке программирования. Команды рисования Работа с графикой.	2	3
	24	Практическое занятие Построение простых графических объектов.	2	3
	Самостоятельная работа обучающихся: 1. Примерная тема доклада: Процессы в объектах Результат: сообщение на уроке по предложенным темам. 2. На оси Oх заданы N точек с координатами x_1, x_2, \dots, x_n . Найти такую точку Z сумма расстояний от которой до данных точек минимальная. Результат: вывод результатов на бумажные носители.		2	3
	<i>Дифференцированный зачет</i>		2	
			Всего:	90

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ДИСЦИПЛИНЫ

3.1. Требования к минимальному материально-техническому обеспечению

Реализация программы дисциплины требует наличия учебного кабинета, Инструментальных средств разработки

Оборудование лаборатории и рабочих мест лаборатории:

Компьютер Р IV –2400-15

Принтер HP-1010

Программное обеспечение WindowsXP, Linux

MsofficeXP, Консультант Плюс, SQL- сервер 2008

1С. Бухгалтерия, FoxPro, Delfi

Электронные учебники: Технология разработки программных продуктов, Базы данных в Delfi, Занимательное программирование, «Делфи»- программирование на языке высокого уровня, Основы алгоритмизации и программирования

3.2. Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Основные источники:

1. Голицына О П, Попов И.И. Основы алгоритмизации и программирования: Учебное пособие.- М.: Форум: Инфра-М, 2014.
2. Семакин И.Г., Шестаков А.П. Основы программирования: Учебник. - М.: Академия 2014.
3. Баженова И. Delphi6: Самоучитель программиста. - М.: КУ-ДИЦ-ОБРАЗ, 2012.
4. Грызлов В.И., Грызлова Т.П. Турбо Паскаль 7.0. - М.: ДМК, 2013.
5. Немнюгин С.А. TurboPascal. - СПб.: Питер, 2015.
6. Фаронов В. Delphi6: Учебный курс. - СПб: Питер, 2016.
7. Фаронов В Турбо Паскаль 7.0 – М: КНОРУС, 2016

Дополнительные источники:

1. Методическая копилка учителя информатики - <http://www.metod-kopilka.ru/>
2. Методическая копилка учителя информатики
<http://nikaschool3.ucoz.ru/index/0-5>

3. Методическая копилка учителя информатики - <http://www.videouroki.net/filecatalog.php>
4. Методическая копилка учителя информатики - <http://86sch6-kogalym.edusite.ru/p48aa1.html>
5. Методическая копилка учителя информатики - <http://comp-science.narod.ru/>
6. Методическая копилка учителя информатики - <http://panoramayrokov.narod.ru/metodik.htm>
7. Интернет университет информационных технологий - <http://www.intuit.ru/>

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ

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

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
В результате освоения дисциплины обучающийся должен уметь:	
<ul style="list-style-type: none"> использовать языки программирования; 	<ul style="list-style-type: none"> - опрос; - тестовый контроль; - выполнение проверочной работы; - оценка результатов выполнения практических работ; - диф.зачет
<ul style="list-style-type: none"> строить логически правильные и эффективные программы; 	<ul style="list-style-type: none"> - опрос; - тестовый контроль; - выполнение проверочной работы; - оценка результатов выполнения практических работ - диф.зачет
<ul style="list-style-type: none"> составлять простые блок-схемы алгоритмов; 	<ul style="list-style-type: none"> - опрос; - тестовый контроль; - выполнение проверочной работы; - оценка результатов выполнения практических работ; - диф.зачет
<ul style="list-style-type: none"> составлять программы на алгоритмическом языке высокого уровня; 	<ul style="list-style-type: none"> - опрос; - тестовый контроль; - выполнение проверочной работы;

	<ul style="list-style-type: none"> - оценка результатов выполнения практических работ; - диф.зачет
<ul style="list-style-type: none"> • работать в интегрированной среде изучаемых языков программирования; 	<ul style="list-style-type: none"> - опрос; - тестовый контроль; -выполнение проверочной работы; - оценка результатов выполнения практических работ; - диф.зачет
<ul style="list-style-type: none"> • взаимодействовать со специалистами смежного профиля при разработке, внедрению и применению объектов профессиональной деятельности; 	<ul style="list-style-type: none"> - опрос; - тестовый контроль; -выполнение проверочной работы; - оценка результатов выполнения практических работ; - диф.зачет
<ul style="list-style-type: none"> • производить модификацию отдельных модулей программы; 	<ul style="list-style-type: none"> - опрос; - тестовый контроль; -выполнение проверочной работы; - оценка результатов выполнения практических работ; - диф.зачет
<ul style="list-style-type: none"> • производить тестирование программного продукта на выявление ошибок. 	<ul style="list-style-type: none"> - опрос; - тестовый контроль; -выполнение проверочной работы; - оценка результатов выполнения практических работ; - диф.зачет
<p>В результате освоения дисциплины обучающийся должен <i>знать</i>:</p>	
<ul style="list-style-type: none"> • общие принципы построения алгоритмов, основные алгоритмические конструкции; 	Индивидуальный опрос на уроках Оценка домашней самостоятельной работ
<ul style="list-style-type: none"> • понятие системы программирования; 	Индивидуальный опрос на уроках Оценка домашней самостоятельной работ
<ul style="list-style-type: none"> • основные элементы процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти; 	Фронтальный опрос по теории Индивидуальный опрос на уроках Оценка деятельности во время практических работ Оценка домашней самостоятельной работы Зачет по теме

<ul style="list-style-type: none"> • подпрограммы, составление библиотек программ; 	<p>Индивидуальный опрос на уроках Оценка деятельности во время практических работ</p>
<ul style="list-style-type: none"> • объектно – ориентированную модель программирования, понятие классов и объектов, их свойств и методов; 	<p>Индивидуальный опрос на уроках</p>
<ul style="list-style-type: none"> • основные приемы программирования. 	<p>Фронтальный опрос по теории Индивидуальный опрос на уроках Оценка деятельности во время практических работ Оценка домашней самостоятельной работы Зачет по теме</p>
<ul style="list-style-type: none"> • интегрированные среды изучаемых языков программирования. 	<p>Индивидуальный опрос на уроках</p>